commodore Magazine

AÑO I - Núm. 5 - Julio 1984 - 250 Ptas.

REVISTA INDEPENDIENTE PARA USUARIOS





Envianos la foto de tu ordenador

En Commodore Magazine hemos pensado que sería buena idea cederos parte del espacio editorial para publicar la foto de vuestro rincón de trabajo. Para ello basta con que nos enviéis cualquier foto en la que se vea, con detalle, como habéis dispuesto vuestra habitación o el comedor de casa. Si preferís aparecer sentados al teclado, tampoco importa. Es conveniente que acompañéis la foto con unas líneas descriptivas de la instalación y, por supuesto, vuestro nombre.

commodore Magazine

Sumario

Commodore Magazine es una publicación de Ediciones y Suscripciones S.A., C/Bravo Murillo, 377 - Madrid 20, Tel. (91) 733 74 13 / 47 / 63 / 97.

REDACCION

Director:
Alejandro Diges.

Colaboradores:
Anibal Pardo.
Gumersindo García.
Roberto Menéndez.
Simeón Cruz.
Miguel Angel de Frutos.
Manuel Arias.

Diseño:
Ricardo Segura.

EDITORIAL

Presidente:
Fernando Bolín.

Director Editorial:
Norberto Gallego.

ADMINISTRACION

Gerente de Circulación y Ventas: Luis Carrero.

Suscripciones: Antonio Zurdo.

Producción:
Miguel Onieva.

Publicidad Madrid:
Roberto Rodriguez.
Bravo Murillo, 377.
Madrid - 20.
Tel. (91) 733 74 13.

Publicidad Barcelona:
Pelayo, 12.
Tel. (93) 301 47 00, Ext. 27

Distribuye: SGEL. Avda. Valdelaparra s/n, Alcobendas, Madrid.

Imprime: Novograph S.A., Ctra. de Irún, Km 12.450 Madrid.

Fotomecánica: Karmat. Pantoja, 10, Madrid.

Depósito Legal: M-6622-1984

Año 1
Num. 5

CONCURSO CALC RESULT

En nuestro próximo número haremos público el resultado del concurso convocado para aplicaciones desarrolladas con el Calc Result.

Esta revista no mantiene relación de dependencia de ningún tipo con respecto de los fabricantes de ordenadores Commodore Business Machines ni de sus representantes.



Por fin llegó el ansiado verano. Es el momento de pasar largas horas delante del ordenador comprado hace pocos meses. Los agobios, las prisas, el tener que abandonar los programas a medias en favor de los libros se acabaron. Por lo menos durante un mes. Deseamos que éste sea el tiempo de sacar mayor provecho y disfrute a **Commodore Magazine** y al ordenador. Imprimir un giro temporal a la línea editorial de la revista se convierte en deuda y obligación durante el período estival. El contenido será principalmente de programas y juegos, así como el **Concurso,** que pasa a tener mayor relevancia. Creemos que también es una buena época para disfrutar de las cinco mil pesetillas ganadas por los felices afortunados.

Hemos aprovechado igualmente para contestar algunas de las cartas que inundan la redacción. Va desde aquí también la firme promesa de intentar responder el mayor número de ellas, que la disponibilidad de espacio nos vaya permitiendo. Desde luego es agobiante dejar fuera algunos de los interesantes artículos elaborados por nuestros colaboradores. Así que, hemos creído que dada la importancia de los temas tratados no importaría demasiado un pequeño retraso en las respuestas.

Este es un buen momento para que os animéis a poner por escrito vuestras experiencias con el **Commodore** y los podáis enviar, en forma de colaboración, a la revista. Tener por seguro que son muchos los usuarios con las mismas inquietudes compartidas, a los que agradará saber cómo habéis resuelto este o aquel problema.

En esta edición continuamos con la serie que trata el tema de cómo diseñar juegos para ordenador. Está próxima a terminar y posteriormente dará paso a un tema que entusiasma a casi todo usuario del **64.**

Incluimos asimismo una breve crónica de la **Quinta Feria Informática Commodore,** organizada en Londres el pasado mes, para aglutinar a todos aquellos que tienen algo que ofrecer a los usuarios, sean accesorios y periféricos, o *software* tanto educativo como profesional y recreativo. Es un acontecimiento al que merece la pena acudir, aunque se trate de un mercado más formado que el nuestro. La recomendación puede ser útil para algunos minoristas, por ser un evento que les puede hacer entender mejor por dónde van, e irán, los tiros. Aclaramos que la Feria tiene carácter anual, pero este año, por primera vez, se volverá a celebrar en la ciudad de Leeds, en septiembre. Si alguien necesita la información, que no dude en contactar con nosotros.

En realidad, por el momento no nos queda otra cosa que desearos un feliz y divertido verano y recordaros que **Commodore Magazine** saldrá también en agosto.

Claves para introducir en el ordenador los programas que aparecen en la revista.

COMO SE VE	COMO SE TECLEA	EFECTO CONSEGUIDO
[illegible]	SHIFT + CLR	(LIMPIA PANTALLA)
[illegible]	CLR	(HOME)
[illegible]	CRSR	(CURSOR ABAJO)
[illegible]	SHIFT + CRSR	(CURSOR ARRIBA)
[illegible]	CRSR	(CURSOR DERECHA)
[illegible]	SHIFT + CRSR	(CURSOR IZQUIERDA)
[illegible]	CTRL + 1	(NEGRO)
[illegible]	CTRL + 2	(BLANCO)
[illegible]	CTRL + 3	(ROJO)
[illegible]	CTRL + 4	(CIAN)
[illegible]	CTRL + 5	(VIOLETA)
[illegible]	CTRL + 6	(VERDE)
[illegible]	CTRL + 7	(AZUL)
[illegible]	CTRL + 8	(AMARILLO)
[illegible]	CTRL + 9	(CARACTER INVERSO)
[illegible]	CTRL + 0	(CARACTER NORMAL)



En esta segunda parte del artículo entramos en los aspectos *soft* del manejo del *interface*.

Si se hubiera cargado ya un programa en el ordenador, se podrá listar mediante LIST en la impresora. Es interesante asegurarse de que funciona con un programa corto. No tardaremos en darnos cuenta de que los símbolos gráficos no aparecen impresos. Basta echar una ojeada al código ASCII para comprender lo que ocurre. En la medida en que los datos son enviados por un solo cable, se hace evidente que la transmisión es realizada en modo serie. Todos los gráficos que aparecen en el teclado del **Vic** son ASCII de ocho bits. Este tipo de símbolos gráficos son una peculiaridad de **Commodore**. Cuando se envían datos a la impresora, entre los cuales se encuentran símbolos gráficos, puede ocurrir una de tres cosas: La impresora ignora el octavo bit, imprimiendo el carácter correspondiente a los restantes siete bits. En tal caso lo que necesitamos enviar son códigos de esta longitud en bits.

Para ordenar al **Vic** que solamente envíe este tipo de códigos es suficiente con cambiar el OPEN por OPEN 2,2,0,CHR$ (39). También la impresora podría ignorar los datos.

Pudieran aparecer otra clase de complicaciones con distintos tipos de impresora, siendo los listados inintelegibles. Si esto ocurre, es necesario invertir los datos transmitidos, haciéndose imprescindible la utilización del registro de Comando. Existe la posibilidad de invertir los datos mediante *software*. En tal caso la transmisión de código ASCII invertido de siete bits a 600 baudios se hace de la siguiente manera: OPEN 2,2,0,CHR$ (224).

Otras impresoras podrían impri-

Interface RS 232 para el VIC-20

mirlo todo, excepto los caracteres gráficos.

Veamos algunos de los problemas que ueden surgir cuando se escribe un programa que proporciona datos de salida por impresora. Consideremos antes las siguientes líneas de programa que es una parte de una rutina de manejo de un gran fichero:

1000 IF Y=1 THEN PRINT #2, A$ SPC (3) B$ SPC(3) C$
1010 IF Y=2 THEN PRINT #, D$ SPC(3) E$ SPC(3) F$
1020 IF Y=3 THEN PRINT #, G$ SPC(3) H$ SPC(3) J$

Una vez introducido y puesto a trabajar (tecleando RUN) funciona sin problema. Posteriormente, se cargaría en cinta, pero al volver a pasarlo desde la cinta al ordenador, aparece el mensaje de SYNTAX ERROR, cuando llega a la línea 300. La causa es simple, falta la variable de retorno del carro, que debería utilizarse al final de cada sentencia PRINT #, R$=CHR$(13), con lo que se evitan todos los problemas.

Se puede necesitar imprimir algo que la mantenga un espacio a partir del borde del papel. Si utilizamos algo como: PRINT #2, "DESDE AQUI", no funcionará. Tampoco pueden ser utilizadas TAB o SPC para espaciar el primer carácter, necesitaremos utilizar otra variable. Utilizando S$= CHR$(0), la línea se escribe: PRINT #2, S$ SPC(6) "DESDE AQUI".

La utilización de CHR$(0) engaña a la impresora, que la primera cosa que demanda es un carácter, puesto que la variable S$ no significa nada, la impresora recibe (felizmente) nada, produciendo el número de espacios en blanco necesarios, indicados por SPC.

Ahora pasemos a ver qué ocurre cuando se necesita volver a utilizar la pantalla. Si hemos abierto el *port* de la impresora en modo directo, no se debe cerrar. El comando CLOSE 2 en este modo es probable que cause problemas. Los programas pueden ser destruidos y el **Vic** quedar colgado, de tal manera que se haga necesario presionar el botón de RESET (puesta a cero), que hemos incorporado al *interface*. Para acceder a la pantalla lo más simple es introducir un carácter (con Return), apareciendo en ella el clásico SYNTAX ERROR, respondiendo la pantalla a

Figura 1. Rutina en código máquina.

```
.,  1200 LDA #$07
.,  1202 STA $1300
.,  1205 LDA #$01
.,  1207 LDX #$00
.,  1209 LDY #$13
.,  120B JSR $FFBD
.,  120E LDA #$02
.,  1210 LDX #$02
.,  1212 LDY #$00
.,  1214 JSR $FFBA
.,  1217 JSR $FFC0
.,  121A LDX #$02
.,  121C JSR $FFC9
.,  121F JMP $6121
```

Forma de inicializar el *port* de impresora y entrar en el **Vicmon:** Se entra en el **Vicmon** con SYS 6*4096, se introduce este programa y se corre con G 1200. El **Vicmon** produce toda su salida por la impresora, lo cual es equivalente a OPEN 2,2,0,CHR$(7): CMD 2: SYS24865.

Figura 2. Rutina de volcado de pantalla.

```
10 OPEN 2,2,0,CHR$(7)
20 G=PEEK(648)*256:FOR P=G TO G+505
30 C=PEEK(P):C$="":IF(P-G)/22=INT((P-G)/22)
THEN PRINT#2,C$
40 IF<32 OR C>95 THEN C=C+64:GOTO 60
50 IF>68 AND C>96 THEN C=C+128
60 C$=C$+CHR$(C)
70 PRINT#",C$;:NEXT
80 RETURN
```

La línea diez normalmente debería situarse al comienzo del programa principal. Las 20 a 80 serán llamadas cuando se requiera.

Figura 3. Rutina de escritura por impresora.

```
10 OPEN 2,2,0,CHR$(7)
20 PRINT"%-F/F @-EXIT ↑-L/F"
30 GET K$:IF K$="" THEN 30
40 IF K$="%"THEN PRINT#2,CHR$(12);:GOTO 30
50 IF K$="↑"THEN PRINT#2,CHR$(10);:GOTO 30
60 IF K$="@"THEN RETURN
70 PRINT K$
80 PRINT#2,K$;
90 GOTO 30
```

Para convertir al **Vic** en una máquina de escribir bórrese Return de la línea 70, reemplazándolo por End.

partir de este momento. Volver a dar salida por impresora es sólo cuestión de teclear CMD2. El cierre del *port* desde el interior de un programa no produce problema alguno.

La manera clásica de entrar en el **Vicmon** es teclear SYS 6*4096. Si abrimos y controlamos el *port* de impresora en BASIC y después entramos en el **Vicmon**, aparecerán todos los registros en la impresora, cerrándose todo lo demás. Durante su rutina de inicialización el **Vicmon** cierra todos los canales de salida sin más miramientos, por tanto también el de la impresora. La forma de entrar en el **Vicmon** y seguir utilizando la impresora la detallamos a continuación:

— Abrir el *port* de impresora desde BASIC.

— Entrar en el **Vicmon** de la manera habitual.

— Cuando se requiera la impresora hay que volver al BASIC.

— Teclear CMD a la impresora.

— Volver a entrar en el **Vicmon** con SYS 24865.

— Presionar nuevamente Return.

Se podría querer obtener un listado en código máquina. Lo más práctico no es intentar correr programas, llegados a este punto. Convendrá salir al BASIC, forzar un SYNTAX ERROR y volver a entrar al **Vicmon.**

Ocurre a menudo que, cuando se sale al BASIC desde el modo impresora, el cursor de pantalla se pierde. Para devolverlo a su forma normal es suficiente con apretar Restores y Run/Stop.

Existe otra manera de trabajar con el **Vicmon** y la impresora. Se abre el canal desde BASIC, se teclea CMD al *port* y se entra en el **Vicmon** con SYS 24865 y Return.

La impresora no ha escrito un '.', pudiéndose putilizar todas las posibilidades del **Vicmon.** Si ensamblamos un programa, cada línea será impresa en cuanto introduzcamos el Return. Si la sintaxis es incorrecta no aparece salida. Cuando quiera correr el programa fuerce un error de sintaxis y vuelva a entrar en el **Vicmon.**

Proporcionamos aquí un programa que inicializa el *port* de impresora y entra en el **Vicmon.** Podría ser utilizado con un mínimo de alteraciones, como rutina de impresión, en un programa mayor. A menudo se emplaza desde la dirección A200 hacia arriba, de tal forma que lo ponemos bastante lejos del resto de la memoria. Resulta obvio que también necesitaremos uno de los cartuchos de RAM direccionables en esta zona (la misma utilizada por los cartuchos de juegos).

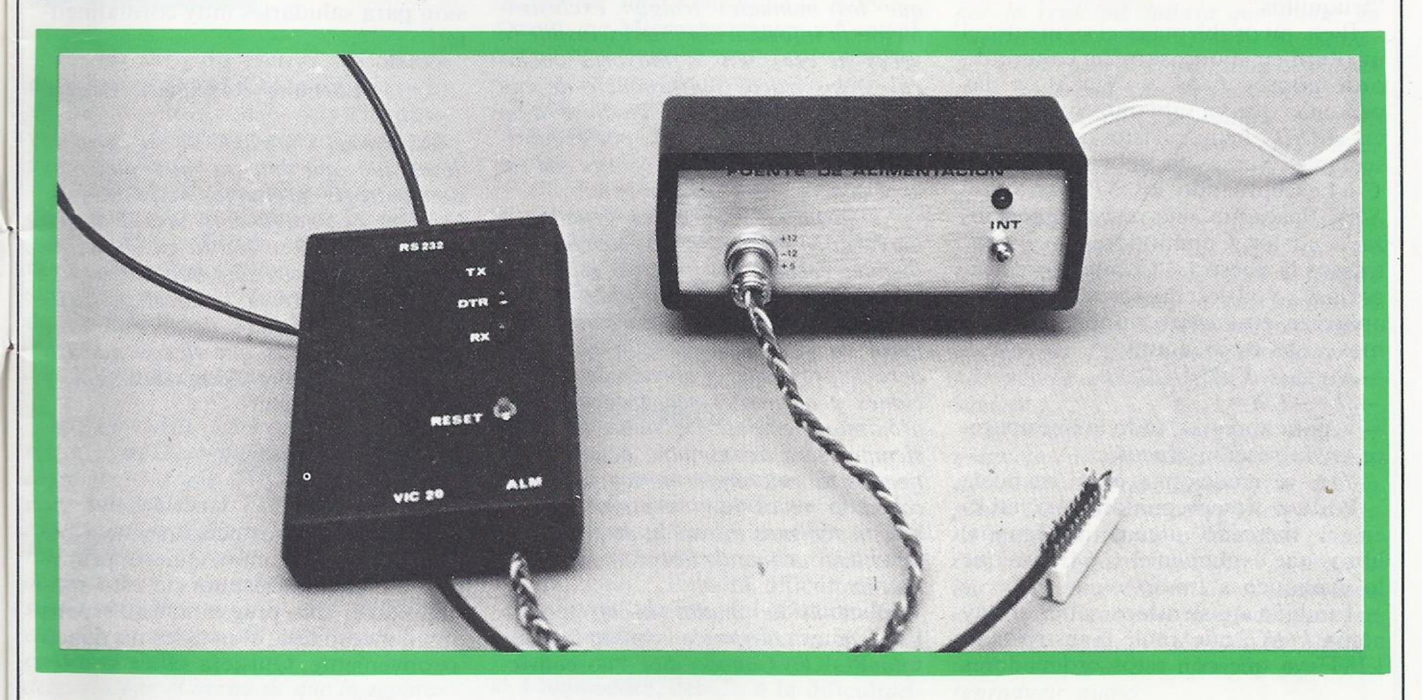
La rutina de volcado de pantalla desde dentro de un programa sirve para obtener una copia directa del contenido que está siendo visualizado. La pantalla del **Vic** sólo tiene 22 caracteres de anchura, pero si desea hacer el volcado, aquí está el programa.

Normalmente, el comando de apertura (OPEN) se sitúa al comienzo del programa, empleándose un GOSUB en otras líneas.

El último programa, para escritura, se autoexplica. Dispone de alimentación de línea y alimentación de hoja. No existe cursor de pantalla.

Para comenzar una nueva línea se teclea ", para alimentar línea y después Return, apareciendo el próximo carácter en la línea siguiente. El programa no ha sido preparado como algo de inusitada utilidad en sí mismo, sino como una pauta para otros desarrollos.

Con esta segunda parte se ha pretendido dejar más claros algunos de los aspectos primarios del funcionamiento de la impresora en conexión con el ordenador. Sin embargo, en un futuro próximo habrá más.



IMPORTANTE:

Son muchas las cartas que diariamente llegan a esta sección. Por tanto rogamos que tengáis un poco de paciencia al esperar la respuesta. Algunas veces las preguntas son similares y quedan respondidas por otra carta que no es la vuestra y se hace obvio que está por demás repetir. Suponemos vuestra comprensión de los cuellos de botella y el funcionamiento en serie del mecanismo de respuesta. Gracias.

Si queréis que hagamos un club *"Vic 20"*, poneros en contacto con: **Antonio Recober.** Tel. 32 20 46 de **Málaga**. Se trata de intercambiar programas, conocimientos y experiencia sin que nos cueste dinero.

Si algún **64** quiere hacer el camino a nuestro lado, podemos intentarlo. ¡Animo malagueños!

A. Recober. Málaga

Ya lo sabéis los "comodoreros" de la zona. Cuando estéis en pleno funcionamiento ya nos contaréis cómo os va.

P.: Ante todo, deseo felicitarles por su revista, una publicación muy bien centrada en los usuarios de los Commodore. Gracias por explicarnos esos "truquillos".

Bien, mi problema es el siguiente: el instituto en el que estoy ha comprado ordenadores C-64 y Vic-20, y las personas que ya sabíamos programar en *BASIC* hemos visto que en dos semanas escasas ya sabemos usar el **C-64** exceptuando los *SPRITES* y el *SYS*. Buscamos una revista especializada en los Commodores, y encontramos la vuestra, "Commodore Magazine", y al leer los programas que aparecen en vuestro número dos hemos visto cosas como:

—*JS=—((X AND 8)=.)*

—*JE=—(X=.)*

—y, para abreviar, todo lo que aparece en la sección trucos.

Y ni se mencionan en el manual.

Por eso nos preguntamos si habrá en el mercado algún (o algunos) libros que expliquen el **C-64**, pero que lo expliquen a fondo.

También quisiéramos saber si hay algún *POKE* que anule la instrucción LIST, ya que con estos ordenadores se pretende dar clase de informática, y no conviene que desde un principio se pongan los alumnos a listar y a copiar programas.

Espero me deis contestación a mis dudas, y que no os moleste si tengo una gran dificultad y os tengo que escribir otra vez. Muchas gracias.

J. Rafael Oscar Martín
Las Palmas de G. C.

R.: *Apreciamos sinceramente la confianza que depositas en nosotros, y puedes escribir las veces que creas necesario. Revistas como esta, y libros también, son el complemento necesario para los usuarios de cualquier marca de ordenador. No se puede pretender que los manuales del fabricante sean excesivamente detallados, pues su negocio es la venta de ordenadores y no el editorial. Por eso es bastante común que los manuales traigan exclusivamente la información imprescinsible (a veces ni eso). En la mayoría de los casos, los trucos son producto de una madurada elaboración, en base a la información suministrada por el fabricante. En otros casos surge de la experiencia, casi casualmente.*

El surtido de libros es grande, la mayoría en idioma inglés, siendo localizables en diversas librerías especializadas, y lógicamente no todas tienen los mismos títulos. La biliografía existente en castellano es aún reducida, pero se prevé que el aluvión de traducciones y originales sea grande en los próximos tiempos. De todas maneras siempre será aconsejable acudir a las tiendas de microinformática más cercanas, a veces hay sorpresas. En la revista tenemos intención de comenzar a publicar crítica de libros no tardando mucho.

El tema de anular la instrucción LIST *no es tan simple como aplicar un solo* POKE. *Cuando dice* **"no conviene"** *interpretamos que se considera no deseable que los alumnos tengan acceso a los listados de los programas didácticos. Hay distintas formas de proteger programas, pero no les quepa la menor duda de que antes o después habrá alguien capaz de dar con la llave. Para indicarle cuál es la forma más efectiva de protección sería deseable conocer alguna característica del programa. Por otro lado, el tipo de protección es dependiente del ordenador, sobre todo cuando afecta a diversas posiciones de memoria. En cualquier caso, conociendo más detalles intentaríamos buscar la solución más factible.*

Desde aquí hacemos una llamada a otros lectores por si tienen problemas parecidos y/o quisieran compartir algunas de las posibles soluciones.

P.: Como nuevo y reciente propietario del ordenador personal **Commodore Vic-20**, desearía recibir información sobre los productos que ustedes pueden ofrecerme.

Sin otro particular y a la espera de sus gratas noticias, aprovecho la ocasión para saludarles muy cordialmente:

Edelmiro Rodríguez. Orense

R.: *Por el contenido de su carta se desprende que piensa que podemos suministrarle productos* hard *y* soft. *Esto no es exactamente así. Nuestra única actividad se centra en el campo editorial, es decir, sólo hacemos revistas. Nuestra oferta se localiza fácilmente en los* **kioskos** *de prensa. Tres son las cabeceras que llevan nuestro sello:* **Commodore Magazine, ZX** *y* **Ordenador Popular.**

P.: Soy un lector de su revista y me gusta su organización. Quiero, primero, felicitarles. Después de esto quisiera hacer una pregunta, que espero obtener respuesa, si ustedes no tienen inconveniente. Quisiera saber la dife-

rencia existente entre el **Vic** sin ampliar y el **Vic** con 16 K de ampliación. Me gustaría además me dijesen donde puedo conseguir un mapa de memoria del **Vic** aquí, en Zaragoza.

Aparte de esto volver a felicitarles y a animarles a continuar, sobre todo con esa parte que dedican al *hardware* en todas las revistas.

J. B. C. Zaragoza

R.: *Recibimos complacidos sus felicitaciones, tal vez inmerecidas. Bien, el principal cambio que ocurre cuando introducimos un cartucho de ampliación de RAM al* **VIC 20** *es que la configuración del mapa de memoria se altera en función del cartucho. El de 3 K no produce variación, pero sí lo hacen el de 8 y 16 Kbytes, que reconfiguran el mapa.*

Para conocer en detalle el mapa de memoria, existen diversos títulos en el mercado, algunos en inglés. Probablemente el más accesible sea la "Guía del Programador para el **Vic 20"**, *escrito en castellano y fácil de encontrar en los comercios de distribución de la marca o en grandes almacenes.*

P.: He dirigido escrito a **Commodore International Ltd.** manifestando mi descontento con el servicio post-venta que la firma tiene en exclusiva para España.

Le adjunto dicho escrito para agradecerle que lo publique en su prestigiosa revista. Espero con ello que tales experiencias como la que me ha sucedido no le ocurran a los usuarios de **Commodore.**

José Antonio García Borrego. Sevilla

R.: *La carta del lector García Borrego viene acompañada de una fotocopia de otra que dirigiera a* **Commodore International Ltd.** *y que por su extensión, no podemos reproducir aquí. En síntesis, se queja en ella de la demora e incertidumbre en que se vio sumido durante más de dos meses de enviada a reparación una unidad de cassette C2N. Expresa también su disgusto por el hecho de que la reparación, más los gastos de envío y llamadas telefónicas que debió hacer para recuperar su material, le saliera por 17.228 pesetas, es decir 5.000 más que el precio de una unidad nueva.*

Como esta es una revista independiente, cuyo único compromiso es con los lectores, nos ha parecido necesario poner en conocimiento de **Microelectrónica y Control** *la acusación del lector García Borrego. Un portavoz autorizado de la empresa nos explicó que, normalmente, el servicio de mantenimiento se hace a través de distribuidores. Al haber optado este lector por el envío directo a Barcelona, se creo una confusión por aparente coincidencia de nombres, que hizo que, una vez efectuada la reparación, la unidad de cassette fuera enviada de vuelta a un distribuidor de otra ciudad. Esta circunstancia explica, según* **Microelectrónica y Control,** *la tardanza de dos meses, hasta que, revisadas todas las entradas y salidas de mercancías, se detectó el error. A la fecha de su carta, 8 de junio, el disgustado usuario ya había obtenido satisfacción a su reclamación. La cuestión del precio de la reparación es tanto o más controvertida. Según los datos que aporta nuestro lector sevillano, la reparación en sí mismo fue facturada a 7.875 pesetas. Precio que* **Microelectrónica y Control** *considera inevitablemente caro teniendo en cuenta la avería de que se trataba. En opinión del importador de* **Commodore,** *el coste de los recambios y de la mano de obra de reparación hacen que el mantenimiento resulte caro si se lo compara con los costes de producción en serie. Problema inherente —añaden— a todo producto de consumo de bajo precio de venta.*

Oidas que fueron las partes, nuestros lectores podrán juzgar.

P.: Quiero comprarme un **Commodore 64,** pero cuando he querido informarme en varias casas de ordenadores personales me han indicado varias pegas, como la dificultad del servicio post-venta del **Commodore.** Por otra parte, he notado una inseguridad en los vendedores con respecto al **Commodore,** debido a la dificultad que he tenido de encontrarlo en las casas de venta.

¿Representa un factor negativo muy a tener en cuenta a la hora de su compra? ¿A qué se debe esta poca fiabilidad de los vendedores? ¿Es quizás a su elevado precio? Quisiera saber también si únicamente el **C-64** funciona con magnetofón **Commodore.**

Félix Baltanas. Castellón

R.: *La primera parte de su carta refleja una inquietud bastante generalizada, cuyo origen es la situación de desabastecimiento que ha sufrido el mercado español (y otros europeos) a partir del último trimestre del año pasado. Efectivamente, hasta hace muy poco, los distribuidores de la marca se quejaban de que el importador no les suministraba equipos para cumplir con la demanda. Esa situación, nos informa* **Microelectrónica y Control,** *ha cambiado y ya no hay razones para temer. En cuanto al servicio post-venta, la misma empresa afirma que no hay pega alguna.*

Ciertamente, si las aprensiones de los distribuidores se vieran refrendadas por la realidad, habría que tomar en cuenta este factor a la hora de comprar un Commodore. Pero, si la situación está superada, no vemos motivo de inquietud. El 64 es un equipo excelente y su fabricante una empresa sólida, líder en el mercado americano de home computers.

En cuanto al precio, ciertamente es comparativamente más alto que en otros mercados europeos, pero éste es un fenómeno común a casi todos los ordenadores personales que se venden en España. En términos absolutos, no nos parece que sea caro, como usted sugiere.

En cuanto a su pregunta sobre el magnetofón, nos parece que podemos darle dos interpretaciones: el **64** *no puede trabajar con un magnetofón que no sea de la marca, a menos de contar con un interface especial. Y, por otra parte, el magnetofón de* **Commodore** *está específicamente diseñado para ese exclusivo uso, por lo que no puede utilizarse para, por ejemplo, grabar o reproducir audio.*

5ª FERIA C

Este año es la quinta ocasión en que se celebra la **Feria Commodore** en las islas británicas. Allí se concentran los principales proveedores de *software* y *hardware* orientado a la marca. **Commodore Magazine** la visitó para poder conocer de primera mano lo que hubo, de tal manera que ahora podemos contarlo sin necesidad de intermediarios.

Es de reconocer que el evento, organizado en el **Novotel** londinense, es capaz de poner los dientes largos a cualquier "comodorero" que se precie. Nada tenemos que decir contra la organización, excelente en todo momento.

Como es lógico, la representación de la central europea de **Commodore Business** fue notable en ambas secciones de la feria. Los exhibidores fueron divididos en dos grandes grupos: los orientados a las aplicaciones domésticas, juegos, etc., y el segmento destinado a las aplicaciones profesionales, de negocio, etc. En el primero, como es lógico, las pantallas con juegos multicoloreados hacían las delicias de los más jóvenes, que destruían marcianos y monstruos sin parar. Por el contrario, en el piso superior, la edad media de los visitantes era superior, como también la gravedad de sus rostros, denotando preocupación por hallar ese nuevo paquete de aplicación, que resuelva las nuevas necesidades de su negocio.

Por lo que a novedades se refiere, había bastantes. Algunas no lo eran tanto para nuestros lectores, tal como el **Commodore 16,** o el **8296**, que ya vimos en la **Feria de Hannover** el pasado abril. Sin embargo, una de las principales atracciones fue el **Plus/4**, que originalmente fue concebido como **264**. La razón aparente de la existencia del modelo parecer ser atribuida a la necesidad de llenar un





OMMODORE

Crónica desde Londres



vacío que actualmente rompe la continuidad entre los modelos profesionales y de uso doméstico en el catálogo de **Commodore.** Para reforzar la idea, el **Plus/4** se anuncia con cuatro paquetes de aplicación, que lo hacen inmediatamente utilizable como herramienta de gestión. Se da la particularidad de que este *software* queda totalmente integrado en la memoria ROM de la máquina.

Los cuatro paquetes cubren cuatro de las áreas claves para las aplicaciones financieras: tratamiento de textos, hoja de trabajo *(spreadsheet)*, base de datos y gráficos orientados a la gestión. Se añade que los cuatro paquetes pueden compartir e intercambiar la información entre sí. Igualmente se ha dotado al **Plus/4** de "facilidades de ventana" en la pantalla, lo cual permite hacer uso simultáneamente de hasta dos paquetes.

Un portavoz de la compañía define al **Plus/4** como "un ordenador personal que permite su utilización como potente herramienta profesional y educativa, al mismo tiempo que una máquina recreativa sofisticada".

Las principales características del modelo se centran en su memoria RAM de 64 Kbytes, de los cuales 60 pueden ser directamente utilizados por los programas desarrollados en BASIC. La memoria ROM inicial tiene una capacidad de 32 Kbytes, en los que se incluye el sistema operativo y el intérprete de la versión avanzada del BASIC, 3.5, que incluye unos 75 comandos, comprendiendo los que afectan a gráficos y sonido. El microprocesador utilizado es el aún no clásico ***7501***. La pantalla utiliza el formato en modo texto de 25 líneas de hasta 40 caracteres cada una, con símbolos, caracteres numéricos y letras mayúsculas y minúsculas. La paleta de colores utilizable es amplia:

121 colores, que en realidad son 15 en ocho niveles distintos de luminancia. Aparte del modo texto están observados los gráficos en alta resolución (200 por 320 puntos) y compartida en modos texto o alta resolución. El teclado lleva incorporadas 67 teclas en total, incluyéndose cuatro para el control del cursor y cuatro de función programables. Para comunicarse con periféricos y accesorios, dispone del mismo tipo de *ports* existentes en el **64**. El precio anunciado para Gran Bretaña es de 250 libras, previendo su disponibilidad para el mes de septiembre.

El modelo **16** es un modelo de bajo precio, que también disfruta del BASIC 3.5, pero solamente lleva 16 Kbytes de RAM. La ROM es igualmente de 32 Kbytes. La definición más aproximada de este modelo sería referirse a él como un **64** con tan sólo 16 K de memoria, de los que 12 K son utilizables por los programas en BASIC.

Uno de los más persistentes rumores que circularon por la feria se refería al **Commodore PC**, parece ser que la multinacional está rediseñándolo con una pantalla de mayor tamaño y mejoras en la CPU. Sin embargo, lo que parece que fue cierto es que los mejores clientes de la firma pudieroan ver el **V8000**, aunque en "petit commite". Se trataría de una potente máquina orientada a su utilización en aplicaciones de gestión.

Otro interesante apartado: las novedades en productos periféricos. En primer lugar podemos destacar la impresora matricial de color **MCS 801**, que con una velocidad de 38 caracteres por segundo es capaz de imprimitir en hasta siete colores. Su compatibilidad con la gama de ordenadores **Commodore** es completa.

La segunda novedad va también de impresoras. En este caso se trata del modelo **DPS 1101**, que trabaja en modo bidireccional y produce documentos en calidad "tipo carta", por lo que su velocidad es lenta en comparación con los modelos matriciales. Imprime a razón de 18 caracteres por segundo y puede utilizar distintos tipos de letra. Su precio de salida es razonable en comparación con modelos de características similares.

El último modelo de impresora es también matricial, no excesivamente rápida (60 caracteres por segundo).

Para almacenamiento masivo se presentaron dos unidades que utilizan diferente tecnología. La primera es la unidad de *diskettes* **SFS 481**, que almacena hasta 170 Kbytes de información, una vez formateado el *diskette*. No representa un incremento en la capacidad con respecto al modelo anteriormente existente. El formato utilizado emplea 35 pistas para grabar los datos, pudiendo definirse hasta 144 ficheros por *diskette*. Como es clásico, la unidad incorpora su propio microprocesador, un **6510**, que libera al ordenador de las tareas propias de controlador de la unidad. El complemento son dos Kbytes de RAM y 16 Kbytes de ROM, conteniendo el sistema operativo para el disco. La velocidad de transferencia de datos sí parece haber sido incrementada, siendo la máxima de 1.675 bytes por segundo.

La otra unidad de almacenamiento masivo es un lector/reproductor de *cassettes*, en realidad es francamente similar a la **C2N**, pero con un conec-



tor diferente, norma DIN, que le hace apto para los **Plus/4 y16.**

El denominador común de todos estos nuevos periféricos es su apariencia externa. En todos ellos se ha empleado el mismo color para el plástico de la carcasa, gris metalizado, exactamente igual que los dos nuevos ordenadores.

Entre las distintas firmas asistentes pudimos ver un poco de todo. Lo que sí queda claro es que el mercado británico ha tenido un enorme desarrollo de cara a los modelos de **Commodore**, particularmente con el **64**, a pesar de la fuerte competencia de los ordenadores de concepción británica.

La firma **Adamsoft** disponía de dos lenguajes de amplia aceptación, el **Tiny FORTH** y el **Zoom Pascal**, en versiones destinadas al **64**.

Procedentes de **Adcomp** y **Adman**, respectivamente, se presentaron una unidad de disco duro y un sintetizador de voz, ambos para el **64**. La firma **Currah** presentó otro sistema de síntesis de voz orientado al **64**, pero con la peculiaridad de traducir texto a voz, es decir, lo que se teclea en el ordenador es "vocalizado" por el dispositivo, llamado **Microspeech 64**.



El material educativo ocupó especial relevancia. Muchas son las casas que disponen de algún tipo de *software*, sea de formación en matemáticas, como el caso de **LCL**, idiomas, o incluso informática. La propia **Commodore** posee algún *software* realmente sorprendente, que por el momento no recordamos haber visto distribuido en nuestro país. Podemos añadir que incluso había alguna firma que oferta cursos de formación para *dealers*.

El tema de los juegos y pasatiempos es un capítulo que merece especial énfasis. Comencemos diciendo que se presentó un nuevo juego deportivo, destinado al **64**. En este caso se trata del baloncesto, cuyas características de presentación en pantalla guardan la línea con las del fútbol, comentado en el pasado número de **Commodore Magazine**. Incluso **Radio Luxemburgo**, que lanzara a la fama a los Beatles, se hizo eco de él, en emisión directa desde la feria. Para añadirle alegría al asunto, un par de jugadores, ataviados con prendas deportivas, aceptaron el desafío de todo aquel dispuesto a puntuar en la canasta.

La empresa **Alligata** disponía de varios juegos emocionantes, especialmente uno que tenía por protagonista a una locomotora, que debe salvar diversas vicisitudes para llegar a su destino. **Anirog Software** presentó cuatro nuevos juegos, cuya calidad gráfica es indiscutible, con **Petch, Ice hunter, Bongo** y **House of Usher**, que hacían las delicias de la concurrencia más joven.

Las mil y una noches también estuvieron presentes en el *Show*. Un excelente juego, con síntesis de voz, basada únicamente en el sintetizador musical estándar del **64** nos lleva a correr mágicas aventuras orientales.

Llamasoft y **Quicksilva**, dos de los más importantes fabricantes de *software* recreativo, también estuvieron presentes con sus *stands*. El primero presentó un atractivo juego, llamado algo así como **La oveja en el espacio.**

Merlin Software presentó un juego aún no disponible comercialmente, pero que lo estará en breve. Habla-

mos de **Wimbledon 84**, un juego de tenis con efecto tridimensional.

Por otro lado, los programas para editar *sprites* en el **64** estaban a la orden del día en bastantes *'stands*.

Los programas de aplicación musical fueron otra de las constantes de la feria, en sus distintas variantes. Es de destacar el **Musicalc**, un sistema destinado a la música creativa, con capacidades realmente sorprendentes. No tardaremos en analizarlo detenidamente en estas páginas, pues este producto de la firma californiana **Waveform** es realmente espléndido.

De cara al diseño gráfico con ordenador pudimos ver diversos accesorios, desde la famosa tableta **Koala Pad**, pasando por el lápiz óptico y una tableta de dibujo con un brazo sin articular, que en su desplazamiento envía las coordenadas instantáneas del bolígrafo al ordenador.

En el área del *software* para aplicaciones profesionales, la oferta también era amplia. Así como la de *hardware* para ampliar las capacidades de los sistemas.

Comencemos por **RAM Electronics**, que ha desarrollado un prototipo de ampliación a 256 Kbytes de RAM para el **64**, e incluso hablan de otra de hasta 512 Kbytes.

APS Microsystems disponía de unidades de almacenamiento masivo, con cartucho móvil, destinadas a los sistemas **8000** y **700**, con capacidades de 10, 20 y hasta 40 Mbytes.

Dataview presentó su red local **Hydra**, capaz de unir diversor **64** para captura de datos, utilizando un **8000** como controlador del conjunto.

Las aplicaciones de contabilidad, como es lógico, proliferaban por toda la exhibición, con todas las variantes que uno pueda imaginar. También había, aunque demasiadas, contabilidades para el presupuesto familiar, tal como el **Cash book 64**, de **Anagram.**

En bases de datos no hubo grandes sorpresas. Tal vez el **Superoffice**, de **Precision Software**, que integra el conocido **Superbase** y el **Superscript II**, con destino al **8096**, era uno de los paquetes más relevantes.

Audiogenic presentó un cartucho, controlado por menú, llamado **Magpie**, capaz para 52 campos y un máximo de unos 3.000 caracteres por registro.

En lo que afecta a las hojas de trabajo, la oferta era variada. Desde el **Visicalc3**, de **Supersoft**, incluyéndose los **busicacl** y sus versiones 2 y 3 (la primera para el **Vic** y las otras dos para el **64**), hasta el interesante **Vizarstar**, de **viza Software**. Dataview tenía el **Insta Calc/Graph**, paquete integrado, compuesto por hoja de trabajo y generación de gráficos. Por lo tocante a tratamiento de textos no encontramos ninguna novedad absoluta. Probablemente, el **Word Result**, de **Handic**, firma que desarrolló el **Calc Result**. En este caso se pueden utilizar ambos paquetes en un **700**, que utilizando las posibilidades de pantalla partida, es capaz de ofrecer en ella simultáneamente el tratamiento de textos y la hoja de trabajo.

Y bien, hasta aquí el resumen de una interesante Feria, que nos ha servido para estar al día en las tendencias del mercado **Commodore**. Es indudable que una marca capaz de generar este tipo de eventos ha logrado una implantación real en el mercado. Veamos cuánto tiempo necesitaremos para poder celebrar algo similar aquí, aunque sea con medios más modestos.



algunos de nuestros programas para los ordenadores personales

commodore











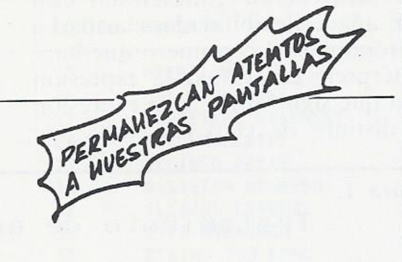












B.M.

BASIC MICRO-ORDENADORES, S.A.

AVD. CESAR AUGUSTO, 72
Tlfs. 23 56 82 y 22 65 44
ZARAGOZA-3

BASIC

El BASIC 4.75 de **Commodore** es el más completo de los aparecidos hasta hoy para los ordenadores **CBM.** Amplía las posibilidades del popular intérprete en algunas facetas importantes, pero manteniendo la compatibilidad con las anteriores versiones del lenguaje. Los códigos binarios de las palabras clave (TOKEN) son idénticos para todas las instrucciones comunes con versiones anteriores (Tabla 1).

La última revisión importante del BASIC para los equipos **Commodore** fue la versión 4.0, vigente desde 1979, y que vino a integrar una serie de comandos orientados, sobre todo, al soporte de los discos y al tratamiento de cadenas.

Ahora, el BASIC 4.75 asume todas las prestaciones del 4.0 y añade novedades en los siguientes aspectos mostrados en la figura 1.

Pasemos a comentar alguna de estas posibilidades.

Tabla 1A.

TOKEN	VERSION 1-3 CLAVE DE PALABRA	TOKEN	VERSION 1-3 CLAVE DE PALABRA
128	END	166	SPC
129	FOR	167	THEN
130	NEXT	168	NOT
131	DATA	169	STEP
132	INPUT≠	170	+
133	INPUT	171	–
134	DIM	172	*
135	READ	173	/
136	LET	174	^
137	GOTO	175	AND
138	RUN	176	OR
139	IF	177	>
140	RESTORE	178	=
141	GOSUB	179	<
142	RETURN	180	SGN
143	REM	181	INT
144	STOP	182	ABS
145	ON	183	USR
146	WAIT	184	FRE
147	LOAD	185	POS
148	SAVE	186	SQR
149	VERIFY	187	RND
150	DEF	188	LOG
151	POKE	189	EXP
152	PRINT≠	190	COS
153	PRINT	191	SIN
154	CONT	192	TAN
155	LIST	193	ATN
156	CLR	194	PEEK
157	CMD	195	LEN
158	SYS	196	STR$
159	OPEN	197	VAL
160	CLOSE	198	ASC
161	GET	199	CHR$
162	NEW	200	LEFT$
163	TAB	201	RIGHT$
164	TO	202	MID$
165	FN	203	GO

IF... THEN... ELSE

Forma general:

IF (expresión); THEN (cláusula then); (:ELSE (cláusula else)).

La bifurcación condicional con ELSE añade flexibilidad y claridad a los programas. Lo primero que hace el intérprete es evaluar la expresión lógica que sigue al IF. Si la expresión vale distinto de cero (es cierta), se

Figura 1.

```
Tratamiento de errores.-                 TRAP   RESUME   DISPOSE
                                         ERR$
Impresión con formato.-                  PUDEF  PRINT USING
Referencias a bancos de memoria.-        BANK
Gestión de cadenas.-                     INSTR
Ficheros.-                               DCLEAR    BLOAD   BSAVE
Teclas programables.-                    KEY
Comandos de manejo de lineas.-           DELETE
Instrucciones de control: Variante       IF...THEN...ELSE
```

Para la serie 700 4.75

ejecutarán la cláusula o cláusulas que siguen al término THEN, hasta encontrar la partícula ELSE o el fin de la línea. La cláusula ELSE se salta, y la ejecución continúa en la línea siguiente. Por el contrario, si la expresión lógica vale cero (es falsa), se ignora cuanto sigue a THEN, y se ejecuta la cláusula o cláusulas que siguen a ELSE hasta encontrar el fin de línea.

La única restricción importante de esta variante de la bifurcación condicional, como de cualquier otra instrucción del BASIC de **Commodore**, es que su longitud no debe superar el número de caracteres permitidos para una línea.

Afortunadamente, la longitud de una línea en la versión que comentamos es de 160 caracteres, a diferencia de las anteriores que era sólo de 80. Ver el ejemplo de la figura 2.

```
10 REM ****** VERIFICA SI X ES UN ENTERO ******
20 IF X=INT(X)
   THEN PRINT X; "es un entero"
   ELSE PRINT X;"no es un entero"
30 IF X=9999 THEN END ELSE 10
```

Figura 2.

Esta pequeña rutina, comprueba si el número introducido por teclado es o no un número entero, comparándolo con el resultado de invocar la función que devuelve la parte entera de un número. El último valor a probar es el número 9999.

Las sentencias IF se pueden anidar o nidificar mientras lo permita la longitud de la línea. Así por ejemplo, podríamos escribir:

IF X>Y THEN PRINT "MAYOR" ELSE IF Y>X THEN PRINT "MENOR";
ELSE PRINT "IGUAL".

Para resolver el clásico problema del tamaño relativo de dos números.

Tratamiento de errores

Las sentencias TRAP, RESUME y DISPOSE son las herramientas que proporciona el BASIC 4.75 de **Commodore** para que el usuario pueda hacer su propio tratamiento de errores durante la ejecución del programa, sustituyendo con sus propias rutinas la actividad normal del intérprete, que generalmente concluye con la emisión de un mensaje y la detención del programa.

La sintaxis de la instrucción TRAP es:

TRAP (número de línea)

"Número de línea" es la dirección de la primera instrucción de la rutina de tratamiento de error especificada por el usuario. Esta rutina puede seleccionar los errores que ha de procesar utilizando las variables reservadas ER, que contiene el número del error de acuerdo a la Tabla 2, y EL, el número de la línea donde está la instrucción que causó el error.

Para pasar de nuevo al tratamiento estándar de errores del BASIC, se pone en el programa una nueva sentencia TRAP, esta vez sin número de línea.

RESUME sirve para conseguir que el programa BASIC prosiga su actividad, una vez detectado y procesado el error. Sin parámetros, este reinicio tiene lugar en la instrucción donde ocurrió el error, y en la forma RESUME NEXT, en la instrucción siguiente a la del error.

Cuando se ejecuta RESUME, vuelve a quedar habilitada la detección y tratamiento por programa de los errores (que no es posible una vez ejecutada una sentencia TRAP), y se inicializan de nuevo las variables ER y EL.

DISPOSE FOR o DISPOSE GOSUB se utilizan también en este contexto de tratamiento de errores, para salir de un bucle FOR o de una subrutina cuando se ha producido un error en ellos y no se desea finalizar su ejecución.

ERR$ (expresión entera) es una variable que contiene el texto del mensaje de error estándar correspondiente a "expresión entera (que debe estar entre 0 y 127) (Tabla 2). La

Tabla 2.

CODIGO	MENSAJE	CODIGO	MENSAJE
0	STOP KEY DETECTED	25	OVERFLOW
1	TOO MANY FILES	26	OUT OF MEMORY
2	FILE OPEN	27	UNDEFINED STATMENT
3	FILE NOT OPEN	28	BAD SUBSCRIPT
4	FILE NOT FOUND	29	REDIM'D ARRAY
5	DEVICE NOT PRESENT	30	DIVISION BY ZERO
6	NOT INPUT FILE	31	ILLEGAL DIRECT
7	NOT OUTPUT FILE	32	TYPE MISMATCH
8	MISSING FILE NAME	33	STRING TOO LONG
9	ILLEGAL DEVICE NUMBER	34	FILE DATA
11	BAD DISK	35	FORMULA TOO COMPLEX
14	BREAK	36	CANNOT CONTINUE
15	EXTRA IGNORED	37	UNDEFINED FUNCTION
16	REDO FORM START	38	LOAD ERROR
20	NEXT WITHOUT FOR	39	VERIFY ERROR
21	SYNTAX ERROR	40	OUT OF STACK
22	RETURN WITHOUT GOSUB	41	UNABLE TO RESUME
23	OUT OF DATA	42	UNABLE TO DISPOSE
24	ILLEGAL QUANTITY		

TOKEN	VERSION 1-3 CLAVE DE PALABRA	VERSION 4 CLAVE DE PALABRA	VERSION 4.75 CLAVE DE PALABRA
204	?SINTAX ERROR(1)	CONCAT	
205		DOPEN	
206		DCLOSE	
207		RECORD	
208		HEADER	
209		COLLECT	
210		BACKUP	
211		COPY	
212		APPEND	
213		DSAVE	
214		CATALOG	
215		RENAME	
216		SCRATCH	
217		DIRECTORY	
218		?SYNTAX ERROR (1)	DCLEAR
219			BANK
220			BLOAD
221			BSAVE
222			KEY
223			DELETE
224			ELSE
225			TRAP
226			RESUME
227			DISPOSE
228			PUDEF
229			USING
230			ERR$
231			INSTR
232			?SYNTAX ERROR (1)

Tabla 1B.

rutina de tratamiento del usuario puede cambiar el texto del mensaje si ello conviene a sus propósitos.

Impresión con formato

El formato y disposición de la salida impresa de un programa cobra una importancia decisiva cuando el ordenador se utiliza para aplicaciones de gestión, como contabilidad, presupuestos, recibos y facturas o declaraciones de impuestos. Con la incorporación de la instrucción PRINT USING, el BASIC de **Commodore** consigue una presentación impresa de números y cadenas de caracteres muy flexible y adaptada casi a cualquier situación. Ciertamente ésta es una de las facetas de la versión 4.5 más prácticas y mejor conseguidas de cara a las aplicaciones corrientes que tienen como resultado un informe impreso (¡qué aplicación informática no lo tiene!), especialmente cuando la salida está destinada a ser utilizada por personal ajeno a la aplicación y generalmente no relacionada con su mecanización.

Este grupo de instrucciones parece pensado para adictos al COBOL (con sus famosas PICTURE o descripciones de formato).

No llega a tanto como él, pero servirá seguramente para despertar las añoranzas (o las fobias) de algún programador, que antes haya estado desarrollando aplicaciones para ordenadores "grandes" (sic).

La sintaxis de la instrucción es:
PRINT USING (formato): (lista).

"Formato" es una colección de símbolos (llamados "campos de formato") que especifican cómo tienen que salir impresos los valores de las variables o expresiones que figuran en "lista". Debe haber tantos campos de formato como valores a imprimir en la lista correspondiente.

Sin entrar a analizar la sintaxis y todos los detalles de la correcta utilización de cada uno de los símbolos que pueden aparecer en un campo de formato, destacaremos algunas de sus posibilidades más interesantes.

Signo monetario: Si se especifica "$" en un campo de formato, el correspondiente valor numérico se imprimirá con el $ delante del primer dígito significativo. Este signo es redefinible.

Signo algebraico: En la primera o en la última posición del campo de formato, puede especificarse el signo: + ó —. Con +, el signo aparece siempre; con —, sólo cuando el número es negativo.

Justificación: El signo "=" en un campo de formato especifica que el valor alfanumérico al que se hace corresponder se imprimirá centrado, y no justificado a la izquierda como ocurriría normalmente. El signo ">" lo justifica a la derecha.

Punto decimal: El punto (.) designa la separación de la parte entera y fraccionaria de un número. Lógicamente sólo puede aparecer una vez en el campo de formato. Este carácter es redefinible.

Comas de separación: Para separar, por ejemplo, grupos de tres cifras en la parte entera. También es redefinible.

Redefinición de caracteres de formato

La instrucción del BASIC 4.75 que permite hacerlo es:
PUDEF (cadena de control).

Los símbolos redefinidos con PUDEF afectarán a todas las PRINT USING que aparezcan más adelante en el programa.

4.75

. En "cadena de control" pueden aparecer de uno a cuatro caracteres. El símbolo que aparezca en primer lugar corresponde al carácter de relleno, que por defecto es el "blanco". El que aparezca en segundo lugar en la "cadena de control" será el que sustituirá al carácter estándar de separación de grupos de tres cifras, que por defecto es la coma (,). Es usual en nuestro entorno, utilizar para esto el punto (.) en lugar de la coma.

En tercera posición viene precisamente el carácter que deseamos represente el punto decimal (por defecto el "."). Los programadores españoles pueden poner aquí la coma (,).

Finalmente, en cuarto lugar, se pone el carácter que se desea aparezca como símbolo monetario ($, por defecto). Lo que desgraciadamente no puede cambiarse, al menos por este procedimiento, es que el símbolo monetario aparezca después y no antes del campo numérico.

La redefinición con PUDEF de estos caracteres no implica que haya que cambiarlos en las especificaciones de formato de PRINT USING. Aquí se emplean siempre los normales, y BASIC sabe, a través de PUDEF, los cambios que habrán de hacerse a la hora de imprimir. Ver el ejemplo de la figura 3.

```
10 REM .....INTERCAMBIO DE COMA Y PUNTO
15 CANT= -1234.567
20 PUDEF " .,"
30 PRINT USING "###,###,###,##";CANT

RUN
-$1.234,57
```

Figura 3.

Notas

1. Cada signo "#" reserva espacio para un dígito.
2. PUDEF ".," respeta el carácter de relleno estándar y no dice nada sobre el signo monetario. En consecuencia estos caracteres no se ven afectados.

Otras posibilidades

BLOAD y BSAVE cargan y salvan respectivamente un fichero binario, a/desde un banco de memoria especificado, DCLEAR inicializa una unidad de disco (lo que en anteriores versiones se hacía abriendo el canal OPEN 15, 8, 15 y escribiendo la orden pertinente PRINT = 15, I(x)).

Finalmente, INSTR es una nueva función intrínseca muy útil para los trabajos con cadenas de caracteres. Realiza la búsqueda de una subcadena en una cadena dada, a partir de un lugar determinado, y devuelve como resultado el número entero correspondiente a la posición donde la encuentra por primera vez (ó 0 si no la halla).

Concurso

Alunizaje

VIC-20

Nada mejor para los aprendices de astronauta que este programa, llamado **ALUNIZAJE**, que recibimos desde **Barcelona** de **Bernardo Berrocal Martí.** Se trata de aterrizar un módulo lunar sobre la superficie de la luna teniendo buen cuidado para no estrellarse, lo que no es nada fácil. Para ello hay que controlar el módulo mediante las teclas **Z, X, C, M** y **N**, tal y como se indica en las instrucciones al comienzo del programa. Además, hay que tener en cuenta que no se puede aterrizar en cualquier parte, sino que hay que hacerlo en una de las varis plataformas previstas para ello. Cada vez que se logra alunizar sin percances, se obtiene como recompensa una cierta cantidad de fuel, además de unos cuantos puntos. Si, por el contrario, no se consigue alunizar, hay que volver a intentarlo una y otra vez hasta que se acabe el fuel, y con él la partida. El módulo inicialmente lleva un cierto impulso hacia la derecha además de la fuerza de atracción de la gravedad lunar, presente en todo momento, y que tira de él hacia abajo. La misión del jugador es contrarrestar todo este conjunto de fuerzas, mediante el teclado del **VIC**, con la habilidad suficiente para conseguir un alunizaje suave.

```
100 GOSUB1000
110 POKE36869,255:POKE36879,8:POKE36878,15
120 PRINT"■■■■■■──────────────────"
130 PRINT"■■■■        ALUNIZAJE "
140 PRINT"■■■■■■──────────────────"
150 PRINT"■■■■■■  Z  X  C   N M"
160 PRINT"■■■■■  !  ";CHR$(34);"  #"
170 PRINT"■■■■■■■■■■■■■■■■■■■■$% ("
190 PRINT"■■■■■  PULSE UNA TECLA   "
195 K=PEEK(197):IFK=64THEN195
200 F=1000:SC=0
210 VX=36:VY=0:X=0:Y=3:P=7724:Q=P:E=3:C=34:V=22::H=3:S=2:T=32:G=1.6
230 GOSUB2000
300 K=PEEK(197):IFK=64THENA=0:B=0:D=32:POKE36877,0:GOTO440
310 IFK=33THENC=33:E=1:V=-1:GOTO360
320 IFK=34THENC=35:E=2:V=1:GOTO360
330 IFK=26THENC=34:E=3:V=22:GOTO360
340 IFK=28THENS=2:H=3:GOTO360
350 IFK=36THENS=4:H=6
360 IFS=2THENPOKE36877,250
370 IFS=4THENPOKE36877,240
400 IFE=1THENA=S:B=0
410 IFE=2THENA=-S:B=0
420 IFE=3THENA=0:B=-S
430 D=C+H
440 IFF>0THEN500
450 A=0:B=0:D=32:POKE36877,0
460 PRINT"■■■■■■■■■■■■■■■■■■■■■■■■SE ACABO EL FUEL";
500 VX=VX+A:VY=VY+G+B
510 X=X+VX+A/2:Y=Y+VY+(G+B)/2:F=F-(ABS(A)+ABS(B))
520 DX=INT(X/400):DY=INT(Y/400)
530 N=P:M=0
540 P=7724+DX+DY*22
550 Q=P+V
560 POKEM,T:POKEN,32:T=PEEK(Q):U=PEEK(P)
565 IFU<>32THEN700
570 POKEQ,D:POKEQ+30720,2:POKEP,C:POKEP+30720,1
580 PRINT"■                     ";
590 PRINT"■■■■H■";INT(VX);TAB(8);"■V■";INT(VY);TAB(15);"■F■";F
600 GOTO300
```

```
700 POKEP,C:POKEP+30720,1:POKE36877,0
710 IFU<>43THEN750
720 IFVY>20THEN740
730 IFVY<5THENPRINT"XXXXXXXXXBUEN ATERRIZAJE";:PO=300:PF=100:GOTO800
740 PRINT"XXXXXXXXX_O LOGRASTE":POKE36876,200:FORY=1TO500:NEXT:POKE36876,0:G
OTO760
750 PRINT"XXXXXXXXXATERRIZAJE ILEGAL":PO=0:PF=0:GOTO780
760 IFVY<10THENPO=200:PF=50:GOTO800
770 PO=100:PF=20:GOTO800
780 POKEP,45:POKEP+30720,2
790 FORI=254TO128STEP-2:POKE36877,I:NEXT:POKE36877,0
800 FORD=1TO1000:NEXT
810 IFP<7944THENPO=PO*3:PF=PF*3:GOTO850
820 IFP<8010THENPO=PO*2:PF=PF*2:GOTO850
830 IFP<8120THEN850
840 PO=PO*5:PF=PF*5
850 PRINT"XXXXXXXPUNTOS +";PO:PRINT"XXXXXXXFUEL +";PF
860 SC=SC+PO:F=F+PF
870 FORD=1TO1000:NEXT
880 IFF>0THEN210
890 IFSC>HSTHENHS=SC
900 PRINT"XXXXXXXXXG A M E  O V E R"
910 PRINT"XXXXXXXXXPUNTUACION X";SC
920 PRINT"XXXXXXXXXRECORD X";HS
940 PRINT"XXXXXXXXXOTRA VEZ ? (Y/N)"
950 K=PEEK(197):IFK=64THEN950
960 IFK=11THEN200
970 POKE36869,240:POKE36879,27:END
1000 FORI=1TO112:READX:POKE7423+I,X:NEXT:RETURN
1010 DATA0,0,0,0,0,0,0,0,142,223,123,219,219,123,223,142
1020 DATA124,254,130,254,124,40,124,146,113,251,222,219,219,222,251,113
1030 DATA1,2,4,8,8,4,2,1,130,68,40,16,0,0,0,0
1040 DATA128,64,32,16,16,32,64,128
1050 DATA3,12,48,192,192,48,12,3,130,130,68,68,40,40,16,16
1060 DATA192,48,12,3,3,12,48,192,255,255,255,255,255,255,255,255
1070 DATA0,0,0,170,85,170,85,170
1080 DATA0,3,103,255,255,255,255,255,137,34,126,255,255,126,36,145
2000 PRINT"XX************************";
2010 PRINT"XX,";SPC(20);",";
2020 PRINT"*,";SPC(18);",*";
2030 PRINT"**";SPC(17);",**";
2040 PRINT"**";SPC(17);"***";
2050 PRINT"**,";SPC(16);"***";
2060 PRINT"***";SPC(16);"***";
2070 PRINT"***       X+X,      ***";
2080 PRINT"***      ***X+X     ***";
2090 PRINT"***.     *XXX<X*      ***";
2100 PRINT"***      *XXX_X*      ***";
2110 PRINT"***      ***  ,X++X,,***";
2120 PRINT"***      ***  ****XX2XX**";
2130 PRINT"***      ***    *****";
2140 PRINT"***      ***     *****";
2150 PRINT"***,     ***     ****";
2160 PRINT"****,X+++X,***      ****";
2170 PRINT"***********,,X++X,,****";
2180 PRINT"*XALUNIZAJEXX****XX5X****";
2190 PRINT"*********************";
2200 POKE8185,42:POKE38905,6:RETURN
```

PROGRAMA ★ commodore Magazine ★ COMPROBADO

Concurso

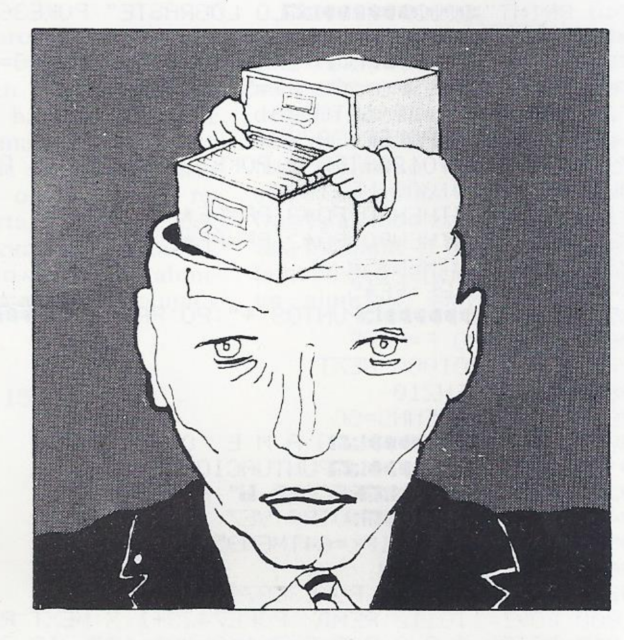
Agenda

AGENDA es un programa para garantizar un fichero con **6** campos: nombre, apellidos, dirección, población, provincia y teléfono. Nos lo remite **Ginés Plaza López**, de **Madrid**. Cada uno de los ficheros se almacena en cinta. Al empezar el programa se puede elegir entre crear un fichero nuevo o cargar uno que estuviera almacenado en cinta. Posteriormente, el programa pasa al menú principal con las opciones de **LOCALIZAR**, **INSERTAR** o **FINALIZAR**.

La opción **LOCALIZAR** permite buscar a través de los registros del fichero y según el contenido de cualquiera de los campos, hasta dar con los datos que se desee. La opción **INSERTAR** permite añadir nuevas fichas al fichero cuando sea necesario, fichas que posteriormente se pueden almacenar en la cinta. Por último, la opción **FINALIZAR** permite salir del programa.

Al elegir la opción **LOCALIZAR** aparece en nuevo menú, con cuatro opciones, para poder llevar a cabo la búsqueda de fichas según el contenido de los diversos campos.

Con este programa y el **CBM 64** ya no hace falta comprar más listines telefónicos.



Commodore 64

```
5 PRINT"□■          A  G  E  N  D  A          "
10 DIM NO$(100),AP$(100),RU$(100)
20 DIM CI$(100),TE$(100),PR$(100)
30 GR=0:N=0
40 PRINT"■■■■DESEA CREAR UN NUEVO FICHERO?  (S/N)"
50 GET D$:IF D$="" THEN 50
60 IF D$="S" THEN 1300
70 IF D$<>"N" THEN 50
80 OPEN 1,1,0,"FICHERO AGENDA"
90 INPUT#1,N
100 FOR I=1 TO N
110 INPUT#1,NO$(I):INPUT#1,AP$(I):INPUT#1,RU$(I)
120 INPUT#1,CI$(I):INPUT#1,TE$(I):INPUT#1,PR$(I)
130 NEXT I
140 CLOSE 1
200 PRINT"□■          A  G  E  N  D  A          "
210 PRINT"■■■     QUE OPERACION DESEA REALIZAR?■■■"
220 PRINT"■          LOCALIZAR........<L>"
230 PRINT"■          INSERTAR.........<I>"
250 PRINT"■          FINALIZAR........<F>"
260 GET D$:IF D$="" THEN 260
270 IF D$="I" THEN 1300
280 IF D$="F" THEN 1400
290 IF D$<>"L" THEN 260
300 PRINT"□■         L O C A L I Z A C I O N         "
310 PRINT"■  POR QUE CLAVE QUIERE BUSCAR?"
320 PRINT"■            NOMBRE:<N>"
330 PRINT"■       APELLIDO(S):<A>"
```

```
340 PRINT"■          CIUDAD:<C>"
350 PRINT"■       PROVINCIA:<P>"
360 PRINT"■■SI DESEA LISTADO COMPLETO,PULSE <T>"
370 GET D$:IF D$="" THEN 370
380 IF D$="N" THEN 510
390 IF D$="A" THEN 1000
400 IF D$="C" THEN 1100
410 IF D$="P" THEN 1200
420 IF D$<>"T" THEN 370
430 PRINT"■■DIRECCIONES Y TELEFONOS.■"
440 FOR I=1 TO N
450 GOSUB 2100
460 NEXT I
470 PRINT"■PARA CONTINUAR, PULSE UNA TECLA"
480 GET D$:IF D$="" THEN 480
490 GOTO 200
510 PRINT"■NOMBRE";:INPUT X$:PRINT"■"
515 SW=0
520 FOR I=1 TO N
530 IF LEN (X$)>LEN(NO$(I)) THEN 550
540 IF LEFT$(NO$(I),LEN(X$))=X$ THEN GOSUB 2100:SW=-1
550 NEXT I
560 IF NOT SW THEN PRINT"NO HAY NADIE CON NOMBRE ";X$:GOTO 470
600 PRINT"QUIERE HACER ALGUN CAMBIO?"
610 GET D$:IF D$="" THEN 610
620 IF D$="N" THEN 200
630 IF D$<>"S" THEN 610
635 GR=-1
640 PRINT"■ACTUALIZACION O BORRADO?"
650 GET D$:IF D$="" THEN 650
660 IF D$="A" THEN 750
670 IF D$<>"B" THEN 650
680 PRINT"■INDIQUE NUMERO DE ELEMENTO A BORRAR";
690 INPUT S
700 IF S>N THEN PRINT"■NO EXISTE.":GOTO 740
710 NO$(S)=NO$(N):AP$(S)=AP$(N):RU$(S)=RU$(N)
720 CI$(S)=CI$(N):TE$(S)=TE$(N):PR$(S)=PR$(N)
730 N=N-1
740 PRINT"■ALGUN CAMBIO MAS?":GOTO 610
750 PRINT"■INDIQUE NUMERO PARA ACTUALIZACION";
760 INPUT S
770 IF S>N THEN PRINT"■NO EXISTE.":GOTO 740
780 PRINT"■SI LO QUE SE MUESTRA ES VALIDO,           ENTRE UNA TECLA."
785 PRINT"EN CASO CONTRARIO, ENTRE EL NUEVO VALOR"
790 PRINT"■NOMBRE: ";NO$(S);
800 INPUT D$:IF LEN(D$)<=1 THEN 820
810 NO$(S)=D$
820 PRINT"APELLIDOS: ";AP$(S);
830 INPUT D$:IF LEN(D$)<=1 THEN 850
840 AP$(S)=D$
850 PRINT"DIRECCION: ";RU$(S);
860 INPUT D$:IF LEN(D$)<=1 THEN 880
870 RU$(S)=D$
880 PRINT"POBLACION: ";CI$(S);
890 INPUT D$:IF LEN(D$)<=1 THEN 910
900 CI$(S)=D$
910 PRINT"PROVINCIA: ";PR$(S);
```

Concurso

Viene de la página anterior

```
920 INPUT D$:IF LEN(D$)<=1 THEN 940
930 PR$(S)=D$
940 PRINT"TELEFONO: ";TE$(S);
950 INPUT D$:IF LEN(D$)<=1 THEN 970
960 TE$(S)=D$
970 GOTO 740
1000 PRINT"■APELLIDO(S)";:INPUT X$:PRINT"⊐"
1010 SW=0
1020 FOR I=1 TO N
1030 IF LEN(X$)>LEN(AP$(I)) THEN 1050
1040 IF LEFT$(AP$(I),LEN(X$))=X$ THEN GOSUB 2100:SW=-1
1050 NEXT I
1060 IF NOT SW THEN PRINT"NO HAY NADIE CON LOS APELLIDOS          ";X$:GOTO 470
1070 GOTO 600
1100 PRINT"■CIUDAD";:INPUT X$:PRINT"⊐"
1110 SW=0
1120 FOR I=1 TO N
1130 IF LEN(X$)>LEN(CI$(I)) THEN 1150
1140 IF LEFT$(CI$(I),LEN(X$))=X$ THEN GOSUB 2100:SW=-1
1150 NEXT I
1160 IF NOT SW THEN PRINT"NO HAY NADIE DE ";X$:GOTO 470
1170 GOTO 600
1200 PRINT"■PROVINCIA";:INPUT X$:PRINT"⊐"
1210 SW=0
1220 FOR I=1 TO N
1230 IF LEN(X$)>LEN(PR$(I)) THEN 1250
1240 IF LEFT$(PR$(I),LEN(X$))=X$ THEN GOSUB 2100:SW=-1
1250 NEXT I
1260 IF NOT SW THEN PRINT"NO HAY NADIE DE LA PROVINCIA ";X$:GOTO 470
1270 GOTO 600
1300 PRINT"⊐■          I N S E R C I O N          "
1305 N=N+1:GR=-1
1310 INPUT"■NOMBRE";NO$(N)
1320 INPUT"■APELLIDOS";AP$(N)
1330 INPUT"■CALLE Y NUM";RU$(N)
1340 INPUT"■POBLACION";CI$(N)
1350 INPUT"■PROVINCIA";PR$(N)
1360 INPUT"■TELEFONO";TE$(N)
1370 GOTO 200
1400 IF GR THEN GOSUB 2000
1410 END
2000 OPEN 1,1,1,"FICHERO AGENDA"
2010 PRINT#1,N
2020 FOR I=1 TO N
2030 PRINT#1,NO$(I):PRINT#1,AP$(I):PRINT#1,RU$(I)
2040 PRINT#1,CI$(I):PRINT#1,TE$(I):PRINT#1,PR$(I)
2050 NEXT I
2060 CLOSE 1
2070 RETURN
2100 PRINT I;":  ";NO$(I);" ";AP$(I)
2110 PRINT TAB(5);RU$(I);" -- ";CI$(I)
2120 PRINT TAB(8);"TFNO: ";TE$(I);"  (";PR$(I);")"
2130 PRINT" ------------------------------------"
2140 RETURN
```

PROGRAMA commodore Magazine COMPROBADO

SU PROGRAMA PARA CUALQUIER SISTEMA COMMODORE PUEDE HACERLE GANAR 5.000 PTAS.

EL PRESENTE CONCURSO ESTA ABIERTO A TODOS NUESTROS LECTORES Y SU PARTICIPACION E INSCRIPCION ES GRATUITA. LEA LAS BASES DEL CONCURSO

- NO SE ESTABLECEN LIMITACIONES EN CUANTO A EXTENSION, TEMA ELEGIDO O MODELO DE ORDENADOR
- LOS CONCURSANTES DEBERAN ENVIARNOS A LA DIRECCION QUE FIGURA AL PIE, EL CASSETTE O DISKETTE CONTENIENDO EL PROGRAMA, UNA EXPLICACION DEL MISMO Y, AL SER POSIBLE, UN LISTADO EN PAPEL DE IMPRESORA. SE PODRAN ENVIAR TANTOS PROGRAMAS COMO SE DESEE
- LOS PROGRAMAS, PREVIA SELECCION, SERAN PUBLICADOS EN LA REVISTA, OBTENIENDO TODOS ELLOS 5.000 PTAS.
- LA DECISION SOBRE LA PUBLICACION O NO DE UN PROGRAMA CORRESPONDE UNICAMENTE AL JURADO NOMBRADO AL EFECTO POR "COMMODORE MAGAZINE", SIENDO SU FALLO INAPELABLE
- LOS CRITERIOS DE SELECCION SE BASARAN EN LA CREATIVIDAD DEL TEMA ELEGIDO Y LA ORIGINALIDAD Y/O SENCILLEZ EN EL METODO DE PROGRAMACION GLOBAL
- ENVIAR A: CONCURSO COMMODORE MAGAZINE



commodore Magazine

C/Bravo Murillo, 377. Madrid -20

Concurso

Mastermind

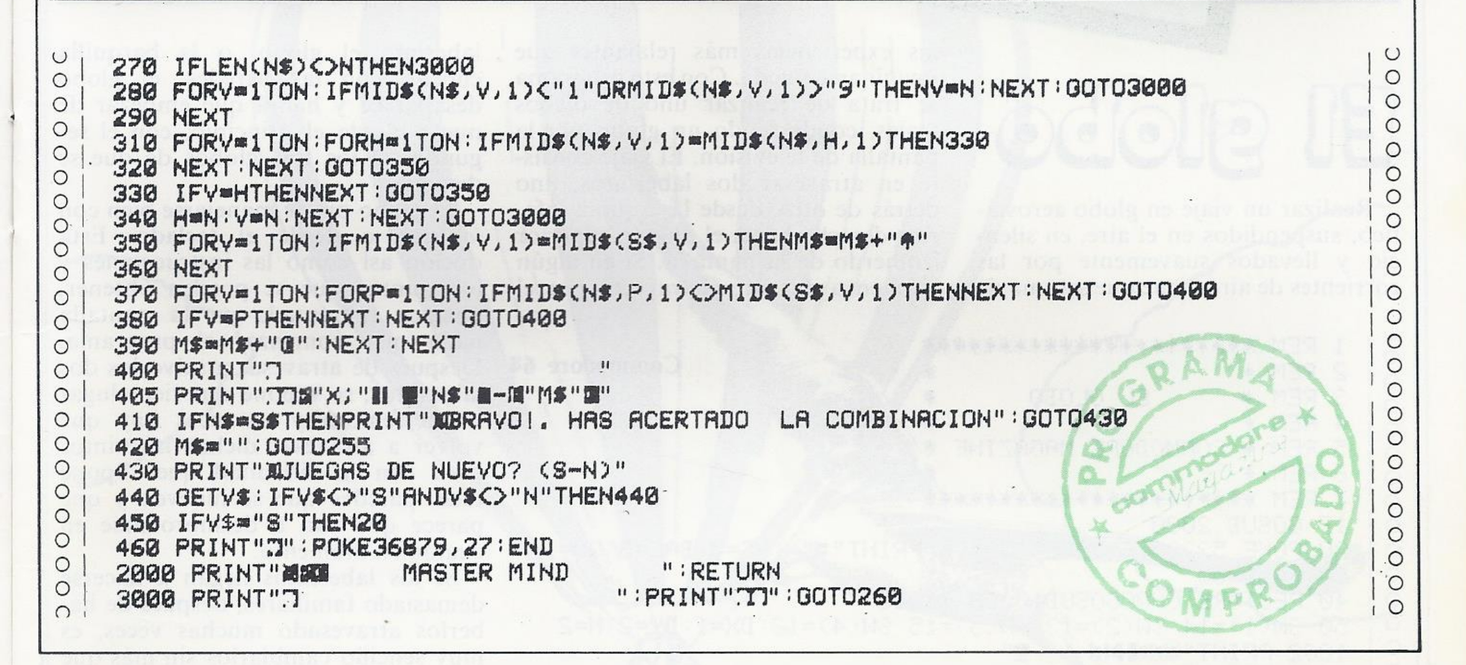
VIC-20

José Antonio García Cortés nos envía, desde Barcelona, el programa MASTERMIND para el **VIC 20**, sin expansión de memoria. Para quien no lo conozca diremos que el Mastermind es un juego entre dos jugadores, en el cual, cada uno de ellos, debe adivinar cierto número secreto pensado por su oponente, con el mínimo número de intentos posible.

En este programa es el ordenador quien genera un número aleatorio, con el número de cifras que elija el jugador al comenzar la partida y sin que ninguna de las cifras aparezca repetida. A partir de aquí el ordenador pedirá que se realice la primera jugada. El jugador tiene que escribir un número, que es analizado por el programa, y que sólo es admitido si cumple con el número de cifras, si no repite ninguna de ellas y si dichas cifras están comprendidas entre uno y nueve, ambas inclusive. Para cada jugada que realice el jugador y que sea admitida como correcta el ordenador la hará figurar en una lista de las jugadas realizadas, y mostrará, al lado de dicha jugada, el balance de aciertos conseguidos. Por cada cifra correcta, que esté en la combinación secreta y en la misma posición que en la jugada, aparecerá un trébol en la pantalla, mientras que por cada cifra presente en la combinación secreta, pero en distinto lugar que en la jugada, aparecerá un carácter @.

Existe un número máximo de 20 intentos para adivinar la combinación, pero si antes de esos 20 intentos el jugador se da por vencido, podrá ver cuál era el número secreto sin más que pulsar la tecla S, después de lo cual, el ordenador preguntará si se desea jugar de nuevo.

```
10 REM MASTER-MIND
20 CLR:PRINT"❐":POKE36879,221:GOSUB2000
30 PRINT"■DEBES ACERTAR LA COM- BINACION SECRETA EN"
40 PRINT"MENOS DE 21 JUGADAS."
50 PRINT"LOS SIGNOS DE INFORMA-CION SOBRE TUS ACIER- TOS SON:"
70 PRINT"  ■■■ : INDICA QUE HAS"
80 PRINT"        ACERTADO UN"
90 PRINT"        NUMERO Y SU"
100 PRINT"        POSICION"
110 PRINT"  ■■■ : INDICA NUMERO"
120 PRINT"        ACERTADO PERO           NO SU POSICION"
130 PRINT"NUMERO DE DIGITOS(2-8)"
135 INPUTN
140 IFN<2ORN>8THEN130
180 PRINT"❐":GOSUB2000
190 FORA=1TON
200 C(A)=INT((RND(1)*9)+1)
210 IFA=1THEN240
220 FORV=1TOA-1:IFC(A)=C(V)THENV=A:NEXT:GOTO200
230 NEXT
240 NEXT:FORV=1TON:H$=STR$(C(V)):C$(V)=RIGHT$(H$,1):NEXT
250 FORV=1TON:S$=S$+C$(V):NEXT
255 X=X+1:IFX=21THENPRINT"■20 JUGADAS. PERDISTE.   SOLUCION: "S$:GOTO430
256 PRINT
260 INPUT"JUGADA";N$
265 IFN$="S"THENPRINT"SOLUCION: "S$:GOTO430
```



```
270 IFLEN(N$)<>NTHEN3000
280 FORV=1TON:IFMID$(N$,V,1)<"1"ORMID$(N$,V,1)>"9"THENV=N:NEXT:GOTO3000
290 NEXT
310 FORV=1TON:FORH=1TON:IFMID$(N$,V,1)=MID$(N$,H,1)THEN330
320 NEXT:NEXT:GOTO350
330 IFV=HTHENNEXT:GOTO350
340 H=N:V=N:NEXT:NEXT:GOTO3000
350 FORV=1TON:IFMID$(N$,V,1)=MID$(S$,V,1)THENM$=M$+"♠"
360 NEXT
370 FORV=1TON:FORP=1TON:IFMID$(N$,P,1)<>MID$(S$,V,1)THENNEXT:NEXT:GOTO400
380 IFV=PTHENNEXT:NEXT:GOTO400
390 M$=M$+"▣":NEXT:NEXT
400 PRINT"▯                                   "
405 PRINT"▯▯▪"X;"▮▮ ▪"N$"▪-▪"M$"▪"
410 IFN$=S$THENPRINT"▪▪BRAVO!. HAS ACERTADO  LA COMBINACION":GOTO430
420 M$="":GOTO255
430 PRINT"▪JUEGAS DE NUEVO? (S-N)"
440 GETV$:IFV$<>"S"ANDV$<>"N"THEN440
450 IFV$="S"THEN20
460 PRINT"▯":POKE36879,27:END
2000 PRINT"▪▪▪     MASTER MIND         ":RETURN
3000 PRINT"▯                                   ":PRINT"▯▯":GOTO260
```

Programas

El globo

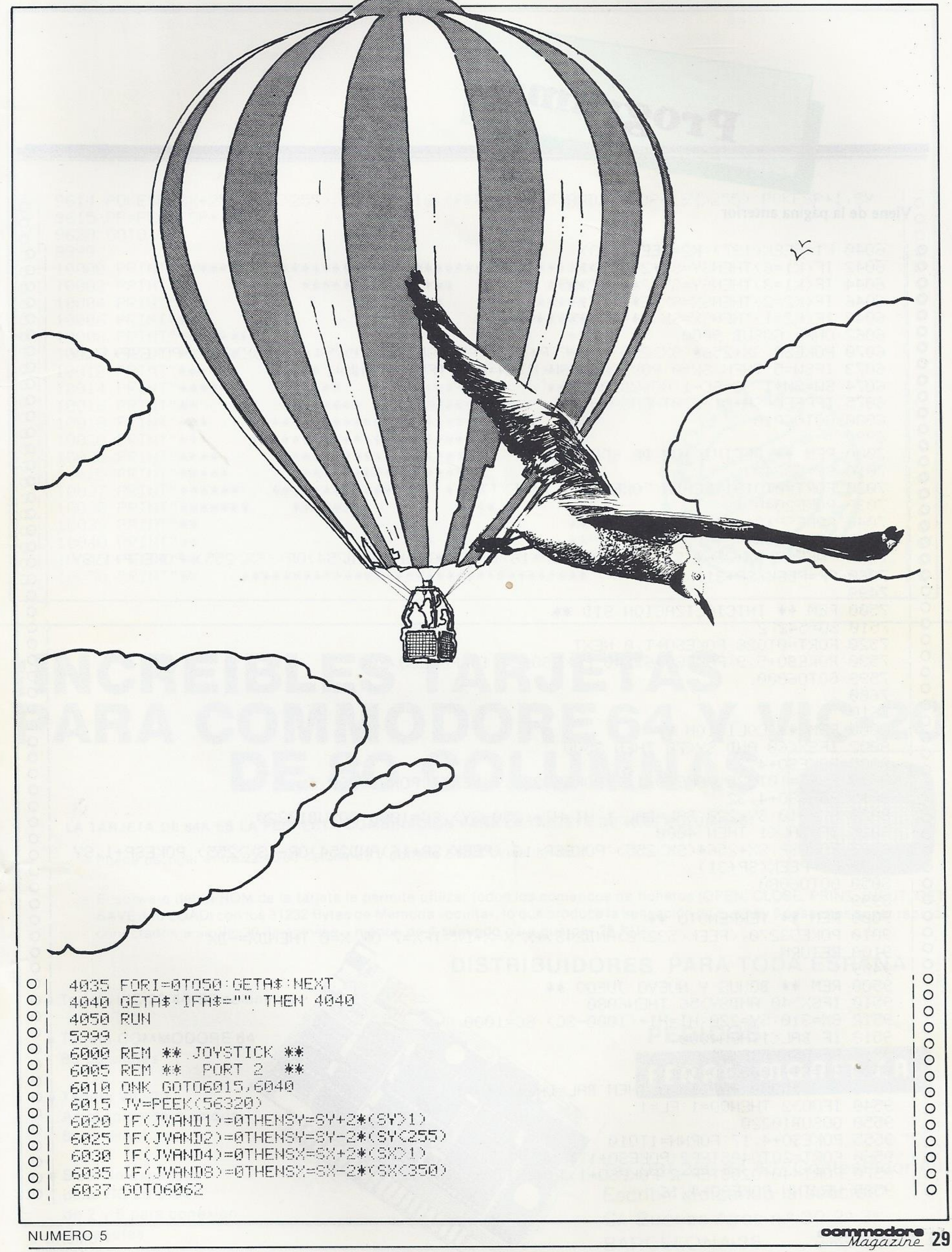
Realizar un viaje en globo aerostático, suspendidos en el aire, en silencio y llevados suavemente por las corrientes de aire, tiene que ser una de las experiencias más relajantes que imaginarse pueda. Con este programa se trata de realizar uno de dichos viajes, conduciendo un globo por la pantalla de televisión. El viaje consiste en atravesar dos laberintos, uno detrás de otro, desde la esquina inferior derecha hasta el ángulo superior izquierdo de la pantalla. Si en algún momento durante el trayecto por el laberinto el globo, o la barquilla chocan con las paredes, el globo desaparece y habrá que empezar de nuevo desde el principio, con el segundo de los tres globos de que se dispone.

El globo puede manejarse bien con *joystick* o desde el teclado. Esta opción así como las instrucciones y los puntos que se pueden obtener, aparece presentada en la pantalla nada más comenzar el programa. Después de atravesar a salvo los dos laberintos, se supone que tiene lugar un terremoto y entonces hay que volver a atravesar dichos laberintos pero con la dificultad que supone unas paredes que se mueven y que parece que van a desmoronarse en cualquier momento.

Si los laberintos llegan a hacerse demasiado familiares, después de haberlos atravesado muchas veces, es muy sencillo cambiarlos sin más que modificar a voluntad las líneas de asteriscos que constituyen las subrutinas de las líneas 10000 y 11000, y que son las que pintan en pantalla los laberintos.

Commodore 64

```
1 REM ***********************
2 REM *                     *
3 REM *      EL GLOBO       *
4 REM *                     *
5 REM * COMMODORE MAGAZINE *
6 REM *                     *
7 REM ***********************
17 GOSUB 2000
20 POKE 53280,6:POKE53281,6:PRINT"■"::HS=0:BAL=3:QQ=1
30 SC=1000
40 RESTORE:ONQQGOSUB10000,11000
50 SW(1)=14:SW(2)=13:SW(3)=15:SW(4)=13:DX=1:DY=2:N=2
1062 PRINT"■■■■■■■        ■"
1063 PRINT"■■■■■■■■1000■"
1064 PRINT"■■■■■■■■■        ■"
1070 GOTO7000
2000 REM ** TITULO **
2010 POKE53280,13:POKE53281,13
2020 PRINT"■■";
2030 PRINT"                    E L  G L O B O"
2040 PRINT"■■■CONDUCE EL GLOBO POR EL LABERINTO CON:"
2045 PRINT
2050 PRINT"                    F 5        "
2052 PRINT"                     !         "
2055 PRINT"                     !         "
2060 PRINT"                CBM--+--SHIFT"
2062 PRINT"                     !         "
2063 PRINT"                     !         "
2065 PRINT"                     !         "
2070 PRINT"                    F 7        "
2075 PRINT
2080 PRINT "TAMBIEN PUEDES IR EN DIAGONAL"
2085 PRINT "POR EJEMPLO CON CBM Y F5 A LA VEZ"
2090 PRINT"■O USAR UN JOYSTICK EN PORT 2■"
2100 PRINT"■GLOBO EXTRA POR ■5000■ PUNTOS"
2110 PRINT"■■ELIJE JOYSTICK ■J■ O TECLADO ■T■"
2120 GETK$:IFK$=""THEN2120
2130 IFK$<>"J"ANDK$<>"T"THEN 2120
2140 IFK$="J" THEN K=1
2150 IFK$="T" THEN K=2
2200 RETURN
2500 :
4000 REM ** FIN DE JUEGO **
4010 PRINT"■■":POKESP+21,0
4015 POKE53280,2:POKE53281,2
4020 PRINT"■■■■■■■■■■"TAB(16)"FIN"
4025 PRINT"■■■"TAB(6)"HAS CONSEGUIDO";HI"PUNTOS"
4030 PRINT"■■■"TAB(10)"■PULSA UNA TECLA■"
```



```
4035 FORI=0TO50:GETA$:NEXT
4040 GETA$:IFA$="" THEN 4040
4050 RUN
5999 :
6000 REM ** JOYSTICK **
6005 REM **  PORT 2   **
6010 ONK GOTO6015,6040
6015 JV=PEEK(56320)
6020 IF(JVAND1)=0THENSY=SY+2*(SY>1)
6025 IF(JVAND2)=0THENSY=SY-2*(SY<255)
6030 IF(JVAND4)=0THENSX=SX+2*(SX>1)
6035 IF(JVAND8)=0THENSX=SX-2*(SX<350)
6037 GOTO6062
```

Programas

Viene de la página anterior

```
6040 K1=PEEK(197):K2=PEEK(653)
6042 IF(K1=6)THENSY=SY+2*(SY>1)
6044 IF(K1=3)THENSY=SY-2*(SY<255)
6046 IF(K2=2)THENSX=SX+2*(SX>1)
6048 IF(K2=1)THENSX=SX-2*(SX<350)
6062 ONFL GOSUB 9000
6070 POKESP,SX+256*(SX>255):POKESP+16,(PEEK(SP+16)AND254)OR-(SX>255):POKESP+1,SY
6073 IFSW=5 THEN SW=0:POKE2040,SW(N):N=N+1:IFN=5THENN=1
6074 SW=SW+1:SC=SC-1:GOSUB10230
6075 IFPEEK(SP+31)<>0THEN8000
6080 GOTO6010
6999 :
7000 REM ** DEFINICION DE SPRITES **
7010 SP=53248
7020 FORT=0TO191:READA:POKE832+T,A:NEXTT
7030 POKE2040,13
7040 POKESP+21,1:POKESP+39,7
7045 SX=310:SY=218
7050 POKESP,SX+256*(SX>255):POKESP+16,(PEEK(SP+16)AND254)OR-(SX>255):POKESP+1,SY
7060 PP=PEEK(SP+31)
7499 :
7500 REM ** INICIALIZACION SID **
7510 SO=54272
7520 FORT=0TO28:POKESO+T,0:NEXT
7530 POKESO+5,9:POKESO+6,240:POKESO+24,15:POKESO,200
7599 GOTO6000
7600 :
7610 :
8000 REM ** COLISION **
8002 IFSX<68 AND SY<70 THEN 9500
8005 POKESO+4,33
8010 FORT=10TO35:POKESO+1,T:POKE53280,T:NEXTT:POKE53280,6
8015 POKESO+4,32
8020 SX=310:SY=220:BAL=BAL-1:HI=HI+(250-SY):SC=1000:GOSUB10220
8025 IF BAL<1 THEN 4000
8027 POKESP,SX+256*(SX>255):POKESP+16,(PEEK(SP+16)AND254)OR-(SX>255):POKESP+1,SY
8030 PP=PEEK(SP+31)
8050 GOTO6080
8999 :
9000 REM ** TERREMOTO **
9010 POKE53270,(PEEK(53270)AND248)+X:X=X+DX:IFX=7 OR X=0 THENDX=-DX
9100 RETURN
9499 :
9500 REM ** BONUS Y NUEVO JUEGO **
9510 IFSX>40 ANDSY>56 THEN6080
9512 SX=310:SY=220:HI=HI+(1000-SC):SC=1000
9513 IF BAL<1THEN4000
9515 PP=PEEK(SP+31)
9520 HI=HI+1000:QQ=QQ+1
9530 IFHI>5000 AND IH=0 THEN BAL=BAL+1:IH=1
9540 IFQQ>2 THENQQ=1:FL=1
9550 GOSUB10220
9555 POKESO+4,17:FORNN=1TO10
9560 FORT=20TO40STEP2:POKESO+1,T:NEXTT
9570 FORT=40TO20STEP-2:POKESO+1,T:NEXTT
9580 NEXTNN:POKESO+4,16
```

```
9614 POKESP,SX+256*(SX>255):POKESP+16,(PEEK(SP+16)AND254)OR-(SX>255):POKESP+1,SY
9615 PP=PEEK(SP+31)
9620 GOTO 40
9999 :
10000 PRINT"□■****************************************";
10002 PRINT"**          ***************         ****";
10004 PRINT"**                ********         *****";
10006 PRINT"**                 ***           *******";
10008 PRINT"********            *                 **";
10010 PRINT"******                                **";
10012 PRINT"******      **                      ****";
10014 PRINT"*****      *****        ******      ****";
10016 PRINT"**        ********     *********     ***";
10018 PRINT"***     *************************     **";
10020 PRINT"**     ***********************        **";
10022 PRINT"****      ********************        **";
10025 PRINT"*****      *******************        **";
10027 PRINT"******   *************    ****        **";
10030 PRINT"*******    ******            **       **";
10035 PRINT"**                                    **";
10040 PRINT"**                     **             **";
10045 PRINT"**                    ****            **";
10050 PRINT"**    **********************************";
```

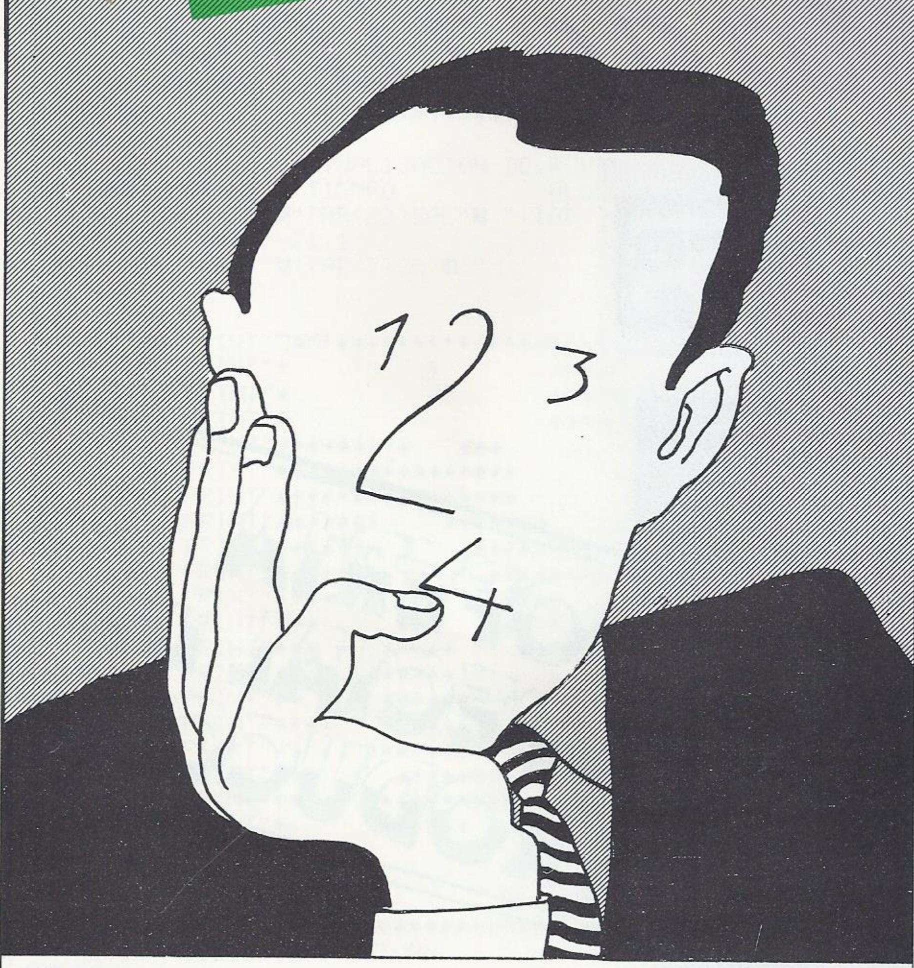
PROGRAMA commodore Magazine COMPROBADO

Programas

Viene de la página anterior

```
10060 PRINT"**                                     **";
10070 PRINT"**                                      *";
10080 PRINT"****                                    *";
10090 PRINT"********************                    *";
10095 PRINT"****************************************";
10199 :
10200 REM ** ACTUALIZACION DE PUNTOS **
10210 PRINT"■ TIEMPO               GLOBOS   PUNTOS"
10220 PRINT"■"TAB(23)BAL"▌ ";TAB(32)HI"▌ "
10225 POKESP+21,1
10230 PRINT"■"TAB(7)SC"▌ "
10290 RETURN
10999 :
11000 PRINT"⌐■■****************************************";
11002 PRINT"*          *        *****         *********";
11004 PRINT"*                  *****            ******";
11006 PRINT"*                  ****              *****";
11008 PRINT"**********   ***    **     *****   ******";
11010 PRINT"****************         *******     *****";
11012 PRINT"****************        **********   *****";
11014 PRINT"*******     *******    ***********    ****";
11016 PRINT"******        ************    ****     ***";
11018 PRINT"*****          ********         *      ***";
11020 PRINT"****    **     *****                ******";
11022 PRINT"***     ***     ****      **          ****";
11025 PRINT"***    *****     **     ******     *******";
11027 PRINT"**    ******           **********     ****";
11030 PRINT"**     ******         *********     ******";
11035 PRINT"****     ******     ************       **";
11040 PRINT"**    *************************     ******";
11045 PRINT"***     ***********************    *******";
11050 PRINT"**     **********      ********     ******";
11060 PRINT"*      ******            ********     ****";
11070 PRINT"***                                      *";
11080 PRINT"*****              ***                   *";
11090 PRINT"******          ********                 *";
11095 PRINT"****************************************■";
11100 GOSUB10200:RETURN
19999 :
20000 REM ** GLOBO 1 **
20002 DATA0,254,0,3,255,128,7,255,192,15,255,224
20003 DATA15,255,224,15,255,224,7,255,192,3,255,128
20004 DATA0,254,0,0,56,0,0,56,0,0,16,0
20005 DATA0,16,0,0,16,0,0,16,0,0,16,0
20006 DATA0,124,0,0,124,0,0,124,0,0,0,0
20007 DATA0,0,0,0
20020 REM ** GLOBO2 **
20022 DATA0,254,0,3,255,128,7,255,192,15,255,224
20023 DATA15,255,224,15,255,224,7,255,192,3,255,128
20024 DATA0,254,0,0,56,0,0,56,0,0,16,0
20025 DATA0,60,0,0,16,0,0,32,0,0,64,0
20026 DATA 12,128,0,63,0,0,31,128,0,15,192,0
20027 DATA7,128,0,0
20100 REM ** GLOBO 3 **
20102 DATA0,254,0,3,255,128,7,255,192,15,255,224
20103 DATA15,255,224,15,255,224,7,255,192,3,255,128
20104 DATA0,254,0,0,56,0,0,56,0,0,16,0
20105 DATA0,60,0,0,8,0,0,4,0,0,2,0
20106 DATA0,1,96,0,0,240,0,3,248,0,3,240
20107 DATA 0,1,224,0
```

Concurso



De todos los programas que pueden escribirse para un ordenador, nadie discutirá la importancia de aquellos orientados a la resolución de problemas matemáticos. Este es el caso de **SISTEMAS DE ECUACIONES,** programa para el **CBM 64,** que recibimos de **Vicente Hortal Puentes,** un lector de nuestra revista en Granada.

El programa resuelve sistemas de ecuaciones lineales, de no importa que número de incógnitas, siempre claro está que el sistema admita solución.

Para utilizar el programa hay que responder en primer lugar a la pregunta "Número de incógnitas?", y si por ejemplo queremos resolver el sistema presentado más abajo (que es un sistema lineal de dos ecuaciones con dos incógnitas) responderemos que 2.

$$X_1 - 2X_2 = -3$$
$$X_1 - X_2 = 2$$

Después hay que introducir los coeficientes de las incógnitas y el término independiente. Se hace por filas, dando lugar a la matriz siguiente tal y como aparece en pantalla.

1 —2 / —3
1 —1 / 2

El programa, después de preguntar si hay algún error que corregir, y al recibir N como respuesta resuelve el sistema, dando como resultado:

$X_1 = 7$
$X_2 = 5$

Sistemas de mayor número de ecuaciones e incógnitas se resuelven con igual sencillez y facilidad.

Sistemas de ecuaciones

Commodore 64

```
10 REM                          PRESENTACION
20 PRINT "▯▯":POKE53280,6:POKE53281,6
30 PRINT "RESOLUCION DE SISTEMAS DE ECUACIONES"
35 PRINT "■LINEALES POR EL METODO DE GAUSS"
40 PRINT "■(ELIMINACIONES SUCESIVAS)                 ------------------------------"
45 PRINT "■■■■ESCRITO PARA COMMODORE-64"
50 PRINT "■POR VICENTE GERMAN HORTAL PUENTES"
55 PRINT "■1# ESCUELAS TECNICAS SUPERIORES"
60 PRINT "■UNIVERSIDAD DE GRANADA"
80 GET D$:IF D$="" THEN 80
90 A$="▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮":C$="                              ":PRINT "▯"
100 REM                     ENTRADA DE DATOS
```

```
110 PRINT A$ C$:INPUT "]NUMERO DE INCOGNITAS ";N:IF N<2 OR INT(N)<>N THEN 110
120 DIM A(N,N+1),R(N+1),X(N):SH=INT(40/(N+2)):IF SH>5 THEN SH=5
130 FOR L=1 TO N
140 FOR C=1 TO N+1:PRINT A$ C$
150 IF C=N+1 THEN PRINT "]TERMINO INDEPENDIENTE "L;:INPUT A(L,C):GOTO170
160 PRINT "]A("L","C")";:INPUT A(L,C)
170 IF N<=8 THEN GOSUB 310
180 NEXT C
190 NEXT L
200 REM                          CORRECCION
210 L=0:IF N<=8 THEN 250
220 L=L+1:PRINT "]███":IF L>N THEN 250
230 FOR C=1 TO N+1:PRINT "A("L","C").=" A(L,C):NEXT C
235 GET D$:IF D$="" THEN 235
240 GOTO 220
250 PRINT A$ C$:INPUT "]ALGUN ERROR ";D$:IF D$="N" THEN PRINT A$ C$:GOTO 410
255 IF D$<>"S" THEN 250
260 PRINT A$ C$:INPUT "]LINEA , COLUMNA ";L,C
270 PRINT A$ C$:PRINT "]A("L","C") ";:INPUT A(L,C)
280 IF N<=8 THEN GOSUB 310
290 GOTO 250
```

Concurso

Viene de la página anterior

```
300 REM          IMPRESION DE COEFICIENTES                    --------------------------
310 PRINT LEFT$(A$,2*L-1);
320 IF C<=N THEN PRINT TAB(SH*C-LEN(STR$(A(L,C)))) A(L,C):GOTO 340
330 PRINT TAB(SH*C-1) "/" TAB(SH*(C+1)-LEN(STR$(A(L,C)))) A(L,C)
340 RETURN
400 REM                   DIAGONALIZACION                                 ----------------
410 FOR P=1 TO N-1:LR=0
420 IF A(P,P)<>0 THEN 500
430 FOR L=P+1 TO N
440 IF A(L,P)<>0 THEN LR=L
450 NEXT L
460 IF LR=0 THEN PRINT "TTTSISTEMA INDETERMINADO":END
470 FOR C=P TO N+1
480 R(C)=A(P,C):A(P,C)=A(LR,C):A(LR,C)=R(C)
490 NEXT C
500 D=A(P,P)
510 FOR C=P TO N+1
520 A(P,C)=A(P,C)/D
530 NEXT C
540 FOR L=P+1 TO N
550 IF A(L,P)=0 THEN 600
560 D=A(L,P)
570 FOR C=P TO N+1
580 A(L,C)=A(L,C)-A(P,C)*D
590 NEXT C
600 NEXT L
610 NEXT P
650 REM                             DISCUSION                                   ----------
660 IF A(N,N)<>0 THEN 710
670 IF A(N,N+1)<>0 THEN PRINT "TTTSISTEMA INCOMPATIBLE":END
680 PRINT "TTTSISTEMA INDETERMINADO":END
700 REM               CALCULO DE INCOGNITAS                    --------------------
710 X(N)=A(N,N+1)/A(N,N)
720 FOR L=N-1 TO 1 STEP -1
730 S=A(L,N+1)
740 FOR C=L+1 TO N
750 S=S-A(L,C)*X(C)
760 NEXT C
770 X(L)=S*(-(ABS(S)>1E-6))
780 NEXT L
800 REM          IMPRESION DEL RESULTADO                     --------------------------
810 PRINT LEFT$(A$,2*N)
820 FOR C=1 TO N
830 PRINT" X"C"="X(C)
840 NEXT C
850 END
```

PROGRAMA COMPROBADO commodore Magazine



Calendario

Commodore 64

Desde Granada, **Pedro Palma Jiménez** nos remite este interesante programa, que lleva por nombre **"Calendario"**. Su función, parece absurdo aclararlo, es la de servir como calendario, ya que presenta cualquier mes de cualquier año, haciendo corresponder los días del mes con el día de la semana correspondiente, tal y como podría encontrarse en un calendario.

Para utilizar el programa hay que copiarlo cuidadosamente en la memoria, después de lo cual, y tras haber escrito RUN, veremos aparecer dos preguntas en la pantalla, correspondientes al mes (entre 1 y 12) y al año que queremos ver. Respondidas estas preguntas, el mes pedido aparecerá en la pantalla.

```
30 CLR:POKE36879,40:PRINT"■":PRINT"□":DIMM$(12):M$(1)="ENERO":M$(2)="FEBRERO"
41 M$(3)="MARZO":M$(4)="ABRIL":M$(5)="MAYO":M$(6)="JUNIO":M$(7)="JULIO":M$(8)="A
GOSTO"
42 M$(9)="SEPTIEMBRE":M$(10)="OCTUBRE":M$(11)="NOVIEMBRE":M$(12)="DICIEMBRE"
60 PRINT"□■■■  CALENDARIO MENSUAL":PRINT"  ==================":PRINT"■■■  DATOS"
62 PRINT"■   SI MES=0 THE END":INPUT"■           MES(1-12)";M:IFM=0THENEND
66 IFM<0ORM>12THEN60
68 PRINT"           _":INPUT"      ANO";Y
69 PRINT"■■■   PARA BORRAR EL CA-   LENDARIO PULSAR -        CUALQUIER TECLA":FORQ
=1TO3000
72 NEXT:PRINT"□■ * MES DE "M$(M)" *":PRINT"          _                    * ANO"Y"*"
80 W=M:D=1:CO=Y/4:RE=Y-INT(CO)*4
100 IFM>2THEN130
110 M=M+12:Y=Y-1
130 N=D+2*M+INT(.6*(M+1))+Y+INT(Y/4)-INT(Y/100)+INT(Y/400)+2
140 N=INT((N/7-INT(N/7))*7+.5)
150 IFN>0THEN180
160 V=7939:X6=23
170 GOTO340
180 IFN>1THEN210
190 V=7942:X7=23
200 GOTO340
210 IFN>2THEN240
220 V=7924:X1=23
230 GOTO340
240 IFN>3THEN270
250 V=7927:X2=23
260 GOTO340
270 IFN>4THEN300
280 V=7930:X3=23
290 GOTO340
300 IFN>5THEN330
310 V=7933:X4=23
320 GOTO340
330 V=7936:X5=23
340 Z=7836:POKEZ,12:POKEZ+3,13:POKEZ+6,24:POKEZ+9,10:POKEZ+12,22:POKEZ+15,19:POK
EZ+18,4
342 POKEV,49:POKEV+3+X7,50:POKEV+6+X6+X7,51:POKEV+9+X5+X7+X6,52
348 POKEV+12+X4+X7+X6+X5,53:POKEV+15+X3+X7+X6+X5+X4,54
352 POKEV+18+X2+X7+X6+X5+X4+X3,55:POKEV+21+X1+X7+X6+X5+X4+X3+X2,56
356 POKEV+24+X7+X7+X6+X5+X4+X3+X2+X1,57:POKEV+26+X6+X7*2+X6+X5+X4+X3+X2+X1,49
360 POKEV+27+X6+X7*2+X6+X5+X4+X3+X2+X1,48
362 POKEV+29+X5+X7*2+X6*2+X5+X4+X3+X2+X1,49:POKEV+30+X5+X7*2+X6*2+X5+X4+X3+X2+X1
,49
366 POKEV+32+X4+X7*2+X6*2+X5*2+X4+X3+X2+X1,49
```

Concurso

Viene de la página anterior

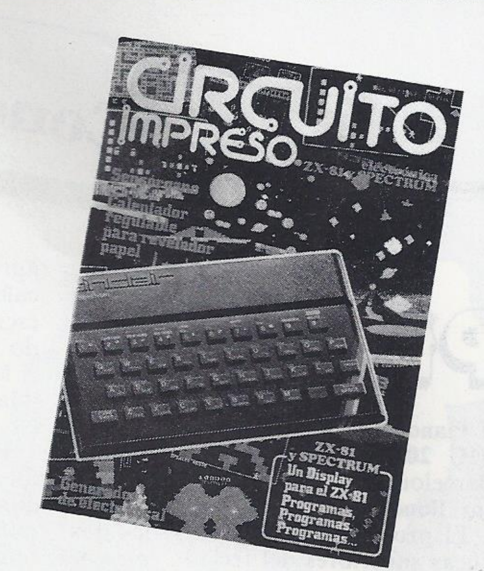
```
368 POKEV+33+X4+X7*2+X6*2+X5*2+X4+X3+X2+X1,50
370 POKEV+35+X3+X7*2+X6*2+X5*2+X4*2+X3+X2+X1,49
372 POKEV+36+X3+X7*2+X6*2+X5*2+X4*2+X3+X2+X1,51
374 POKEV+38+X2+X7*2+X6*2+X5*2+X4*2+X3*2+X2+X1,49
376 POKEV+39+X2+X7*2+X6*2+X5*2+X4*2+X3*2+X2+X1,52
378 POKEV+41+X1+X7*2+X6*2+X5*2+X4*2+X3*2+X2*2+X1,49
380 POKEV+42+X1+X7*2+X6*2+X5*2+X4*2+X3*2+X2*2+X1,53
382 POKEV+44+X7+X7*2+X6*2+X5*2+X4*2+X3*2+X2*2+X1*2,49
384 POKEV+45+X7+X7*2+X6*2+X5*2+X4*2+X3*2+X2*2+X1*2,54
386 POKEV+47+X6+X7*3+X6*2+X5*2+X4*2+X3*2+X2*2+X1*2,49
388 POKEV+48+X6+X7*3+X6*2+X5*2+X4*2+X3*2+X2*2+X1*2,55
390 POKEV+50+X5+X7*3+X6*3+X5*2+X4*2+X3*2+X2*2+X1*2,49
392 POKEV+51+X5+X7*3+X6*3+X5*2+X4*2+X3*2+X2*2+X1*2,56
394 POKEV+53+X4+X7*3+X6*3+X5*3+X4*2+X3*2+X2*2+X1*2,49
396 POKEV+54+X4+X7*3+X6*3+X5*3+X4*2+X3*2+X2*2+X1*2,57
398 POKEV+56+X3+X7*3+X6*3+X5*3+X4*3+X3*2+X2*2+X1*2,50
400 POKEV+57+X3+X7*3+X6*3+X5*3+X4*3+X3*2+X2*2+X1*2,48
402 POKEV+59+X2+X7*3+X6*3+X5*3+X4*3+X3*3+X2*2+X1*2,50
404 POKEV+60+X2+X7*3+X6*3+X5*3+X4*3+X3*3+X2*2+X1*2,49
406 POKEV+62+X1+X7*3+X6*3+X5*3+X4*3+X3*3+X2*3+X1*2,50
408 POKEV+63+X1+X7*3+X6*3+X5*3+X4*3+X3*3+X2*3+X1*2,50
410 POKEV+65+X7+X7*3+X6*3+X5*3+X4*3+X3*3+X2*3+X1*3,50
412 POKEV+66+X7+X7*3+X6*3+X5*3+X4*3+X3*3+X2*3+X1*3,51
414 POKEV+68+X6+X7*4+X6*3+X5*3+X4*3+X3*3+X2*3+X1*3,50
416 POKEV+69+X6+X7*4+X6*3+X5*3+X4*3+X3*3+X2*3+X1*3,52
418 POKEV+71+X5+X7*4+X6*4+X5*3+X4*3+X3*3+X2*3+X1*3,50
420 POKEV+72+X5+X7*4+X6*4+X5*3+X4*3+X3*3+X2*3+X1*3,53
422 POKEV+74+X4+X7*4+X6*4+X5*4+X4*3+X3*3+X2*3+X1*3,50
424 POKEV+75+X4+X7*4+X6*4+X5*4+X4*3+X3*3+X2*3+X1*3,54
426 POKEV+77+X3+X7*4+X6*4+X5*4+X4*4+X3*3+X2*3+X1*3,50
428 POKEV+78+X3+X7*4+X6*4+X5*4+X4*4+X3*3+X2*3+X1*3,55
430 POKEV+80+X2+X7*4+X6*4+X5*4+X4*4+X3*4+X2*3+X1*3,50
432 POKEV+81+X2+X7*4+X6*4+X5*4+X4*4+X3*4+X2*3+X1*3,56
434 IFRE<>0ANDW=2THEN500
436 POKEV+83+X1+X7*4+X6*4+X5*4+X4*4+X3*4+X2*4+X1*3,50
438 POKEV+84+X1+X7*4+X6*4+X5*4+X4*4+X3*4+X2*4+X1*3,57
440 IFW=2THEN500
442 POKEV+86+X7+X7*4+X6*4+X5*4+X4*4+X3*4+X2*4+X1*4,51
444 POKEV+87+X7+X7*4+X6*4+X5*4+X4*4+X3*4+X2*4+X1*4,48
446 IFW=4ORW=6ORW=9ORW=11THEN500
448 POKEV+89+X6+X7*5+X6*4+X5*4+X4*4+X3*4+X2*4+X1*4,51
450 POKEV+90+X6+X7*5+X6*4+X5*4+X4*4+X3*4+X2*4+X1*4,49
500 CLR:GETA$:IFA$=""THEN500
510 GOTO30
```





ANTES ÉRAMOS

CIRCUITO IMPRESO



* **

AHORA SOMOS

CIRCUITOS & COMPUTADORAS



...y estamos
en la calle
a disposición
de nuestros lectores

Concurso

Piano

Piano es un programa destinado al **VIC 20** que nos ha llegado desde Barcelona de manos de **Xavier Escobar Bonet,** lector de nuestra revista.

El programa consigue que las dos líneas superiores del teclado se comporten como teclas de un piano, aprovechando las capacidades musicales del **VIC 20.** Para ello basta escribir RUN después de haber copiado cuidadosamente el programa.

Más interesante quizás es la posibilidad de almacenar canciones tanto en memoria como en un *cassette.*

Para almacenar en memoria, hay que pulsar la tecla F1, nada más comenzar el programa, pasando después a tocar la canción. Cuando se ha terminado de tocar, hay que pulsar F5. Ahora la canción está grabada en memoria y para escucharla de nuevo basta con pulsar F3, que ella sola irá saliendo, nota a nota por el altavoz.

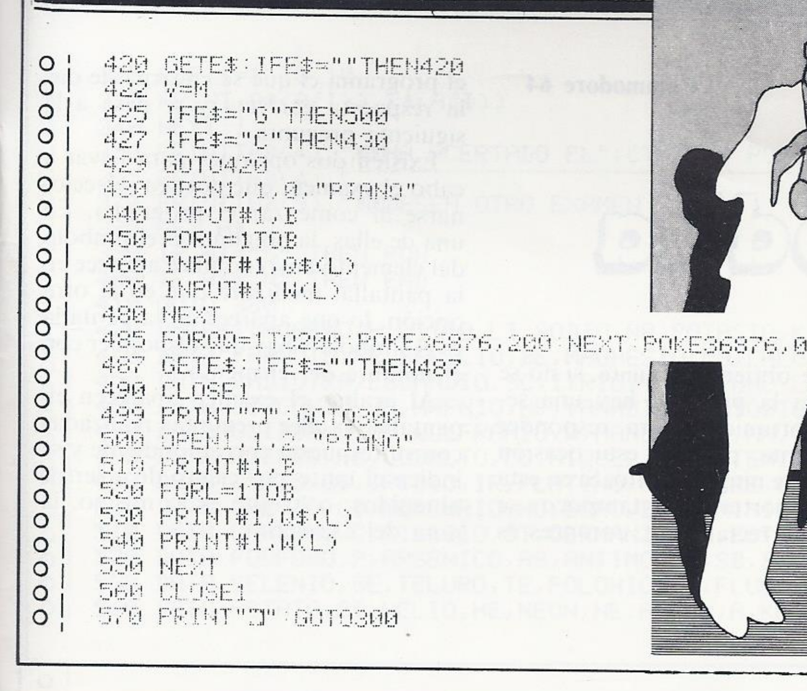
Para almacenar la canción en *cassette* el procedimiento es el siguiente: Después de grabarla en memoria como se ha explicado se debe pulsar la tecla F7, hecho lo cual el **VIC 20** preguntará "GRABAR O CARGAR?".

Escribiendo G como respuesta, el programa pasa a grabar la canción en la cinta. Si la respuesta es C entonces el programa carga desde la cinta una canción en el ordenador, canción que puede escucharse pulsando como antes la tecla F3.

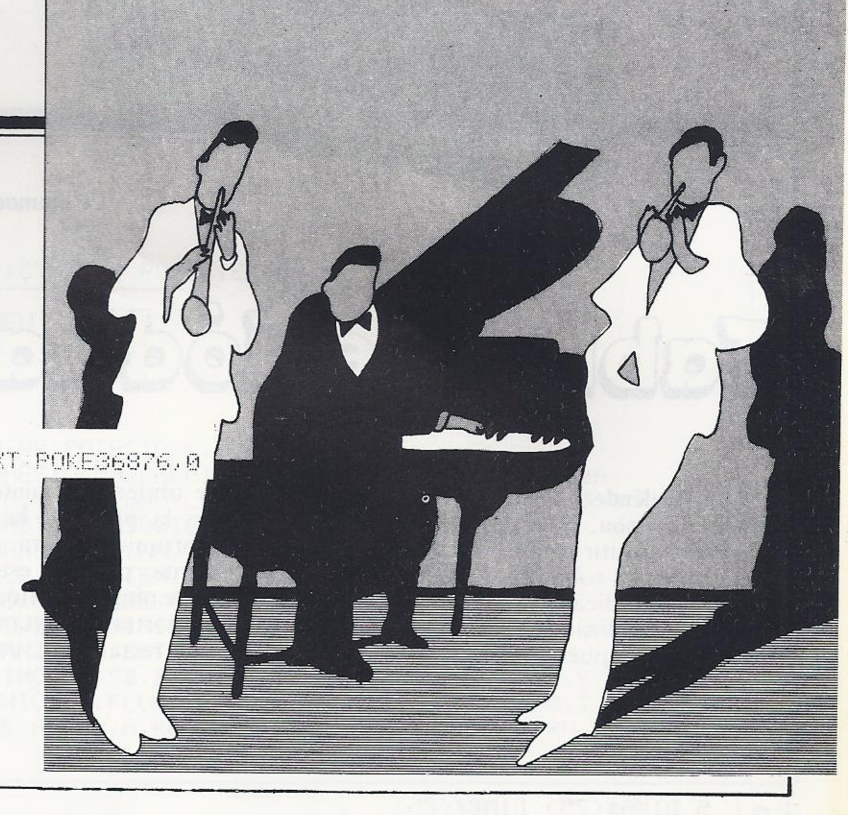
VIC-20

```
1 PRINT"": B=1
10 DATA235,0,0,233,0,0,229,191,199,203
20 DATA0,212,217,221,0,227,0,0,0,0
30 DATA0,0,232,0,0,0,0,207,0,0
40 DATA0,225,0,0,0,0,0,228,231,195
50 DATA209,0,215,223,0,201,0,219,0,0
60 DATA236,0,237
70 DATA15,11, 7,6,5,5,4,4,4,3,3,3,2,2,2,2,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0
100 DIM N(53),D(39),Q$(200),W(200)
110 FORJ=1TO53:READN(J):NEXT
120 FORJ=1TO39:READD(J):NEXT
130 FORJ=1TO4
140 POKE36873+J,0
150 NEXT
160 T=40
165 REM----------------
170 IF T<40THEN POKE 36878,D(T):T=T+.5
200 GETA$:IFA$<>""THEN 220
205 IFQ=1THENO=O+1
210 GOTO170
220 :
221 IFQ=1THENW(M+1)=O:M=M+1:O=0
222 IFQ=1THENQ$(B)=A$
225 IFA$="■"THENQ=1:A$="4":Q$(1)="4":B=1:M=0
227 IFA$="■"THENPOKE36876,0:POKE36875,0:GOTO300
228 IFA$="■"THENQ=0
229 IFA$="■"THEN400
230 A=ASC(A$)-41
240 IFA<1 ORA>53THEN200
245 IFQ=1THENB=B+1
250 POKE36875,N(A):POKE36876,N(A):T=1:GOTO170
300 Z=0
305 POKE36878,15
310 FORZ=1TOB-1
320 S=ASC(Q$(Z))-41
322 FORP=1TOINT(W(Z))
323 IF T<40THEN POKE 36878,D(T):T=T+.5
325 POKE36876,N(S):POKE36875,N(S):NEXT
326 T=1
327 NEXT
330 POKE36878,0
340 GOTO200
400 :
410 PRINT"GRAVAR O CARGAR?(G/R)"
```

PREMIADO CON **5.000** PESETAS



```
420 GETE$:IFE$=""THEN420
422 V=M
425 IFE$="G"THEN500
427 IFE$="C"THEN430
429 GOTO420
430 OPEN1,1,0,"PIANO"
440 INPUT#1,B
450 FORL=1TOB
460 INPUT#1,Q$(L)
470 INPUT#1,W(L)
480 NEXT
485 FORQQ=1TO200:POKE36876,200:NEXT:POKE36876,0
487 GETE$:IFE$=""THEN487
490 CLOSE1
499 PRINT"J":GOTO300
500 OPEN1,1,2,"PIANO"
510 PRINT#1,B
520 FORL=1TOB
530 PRINT#1,Q$(L)
540 PRINT#1,W(L)
550 NEXT
560 CLOSE1
570 PRINT"J":GOTO300
```



Concurso

Commodore 64

Tabla periódica

Luis Meléndez Aganzo nos remite, desde **Córdoba,** este programa que nos va a permitir comprobar nuestros conocimientos sobre los elementos de la tabla periódica. El programa hace el examen realizando **diez** preguntas. Por cada respuesta correcta a la primera se obtiene **un punto,** si no se responde a la primera, hay una segunda oportunidad para responder correctamente, pero en esta ocasión no se obtiene ningún punto. Si en esta segunda oportunidad tampoco se responde correctamente, entonces es el programa el que se encarga de dar la respuesta correcta y se pasa a la siguiente pregunta.

Existen dos opciones para llevar a cabo el examen, que pueden seleccionarse al comenzar el programa. En una de ellas, la pregunta es el símbolo del elemento cuyo nombre aparece en la pantalla, mediante que en la otra opción, lo que aparece en la pantalla es el símbolo y hay que responder con el nombre del elemento.

Al acabar el examen aparecen en pantalla las **diez** preguntas realizadas con su respuesta correspondiente y se indica el tanto por ciento de aciertos obtenidos, o lo que es lo mismo, la nota del examen.

```
5 DIMA$(75):DIMB$(75)
7 PRINT"■"
10 POKE53280,7:POKE53281,2
15 PRINT"■■     T A B L A     P E R I O D I C A     "
30 PRINTTAB(3);"■■■■QUE QUIERES QUE TE PREGUNTE :"
35 PRINTTAB(3);"────────────────────────────────"
40 PRINTTAB(3);"■1.-EL SIMBOLO DADO EL ELEMENTO"
50 PRINTTAB(3);"■2.-EL ELEMENTO DADO EL SIMBOLO"
60 PRINTTAB(5);"■■PULSA ■1■ O ■2■ SEGUN DESEES"
70 GET A$:IF A$<>"1" AND A$<>"2"THEN 70
80 PRINT"■"
100 FOR I=1 TO 10
110 A(I)=INT(75*RND(1))+1
120 IF I=1 THEN 160
130 FOR J=1 TO I-1
140 IF A(I)=A(J) THEN 110
150 NEXT J
160 NEXT I
170 FOR I=1 TO 75
180 READ A$(I),B$(I)
190 NEXT I
200 FOR I=1 TO 10
210 IF A$="1" THEN 230
220 PR$=B$(A(I)):SO$=A$(A(I)):GOTO 240
230 PR$=A$(A(I)):SO$=B$(A(I))
240 PRINT PR$,
250 INPUT B$:IF B$=SO$ THEN 300
260 PRINT"■■NO ES CORRECTO■.INTENTALO DE NUEVO"
270 INPUT Z$:IF Z$=SO$ THEN 310
280 PRINT"■LA RESPUESTA CORRECTA ES:■";SO$;"■"
290 GOTO 310
300 CT=CT+1
310 PRINT"----------------------------------------"
315 NEXT I
320 FOR I=1 TO 500:NEXT I:PRINT"■"
```

```
325 FOR I=1 TO 10
330 PRINT A$(A(I)),B$(A(I))
340 NEXT I
350 PRINTTAB(4);"QQHA ACERTADO EL";CT*10;" POR CIENTO"
355 PRINTTAB(4);"________________________"
370 PRINTTAB(9);"QQDESEA OTRO EXAMEN ?"
380 GET A$
390 IF A$="S" THEN RUN
400 IF A$="N" THEN END
410 GOTO 380
430 DATAHIDROGENO,H,LITIO,LI,SODIO,NA,POTASIO,K,RUBIDIO,RB,CESIO,CS
440 DATA FRANCIO,FR,BERILIO,BE,MAGNESIO,MG,CALCIO,CA,ESTRONCIO,SR,BARIO,BA
450 DATA RADIO,RA,ESCANDIO,SC,ITRIO,Y,LANTANO,LA,ACTINIO,AC,TITANIO,TI
460 DATA ZIRCONIO,ZR,HAFNIO,HF,VANADIO,V,NIOBIO,NB,TANTALO,TA,CROMO,CR
470 DATAMOLIBDENO,MO,WOLFRAMIO,W,MANGANESO,MN,TECNECIO,TC,RENIO,RE
480 DATA HIERRO,FE,COBALTO,CO,NIQUEL,NI,RUTENIO,RU,RODIO,RH,PALADIO,PD
490 DATA OSMIO,OS,IRIDIO,IR,PLATINO,PT,COBRE,CU,PLATA,AG,ORO,AU,CINC,ZN
500 DATA CADMIO,CD,MERCURIO,HG,BORO,B,ALUMINIO,AL,GALIO,GA,INDIO,IN,TALIO,TL
510 DATA CARBONO,C,SILICIO,SI,GERMANIO,GE,ESTANO,SN,PLOMO,PB,NITROGENO,N
520 DATA FOSFORO,P,ARSENICO,AS,ANTIMONIO,SB,BISMUTO,BI,OXIGENO,O,AZUFRE,S
530 DATA SELENIO,SE,TELURO,TE,POLONIO,PO,FLUOR,F,CLORO,CL,BROMO,BR,YODO,I
540 DATA ASTATO,AT,HELIO,HE,NEON,NE,ARGON,A,KRIPTON,KR,XENON,XE,RADON,RN
```

Programas



Piedras a los topos

Este programa quizá no sea el más adecuado para los amantes de los animales, pues consiste en lanzar piedras contra unos graciosos topos, que intentan, abriendo el terreno, sacar sus cabecillas al aire. Pero como no deja de ser un juego, y en realidad no hay animosidad contra los pequeños topos, creemos que todos disfrutarán jugángolo.

El jugador dispone de 15 piedras para lanzar. Hay tres montículos bajo los cuales trabajan los topos abriendo una vía al exterior. Para dar a los topos hay, primero, que destruir las defensas que los protegen, lo que se consigue lanzando las piedras a los costados de los montículos. Como los animalitos están constantemente subiendo y bajando la cabeza, aunque la piedra llegue al lugar adecuado, puede fallar si el topo baja la cabeza, es preciso darles justo cuando están arriba. Cuando se acierta, aparece un destello rojo, se escucha un "splatch" y el jugador se apunta 10 puntos. Si falla, no se apunta ningún punto, pero sigue habiendo ruidos cuando la piedra llega al suelo.

Tanto si se acaban las bombas, como si transcurre el tiempo disponible, aparecen los mensajes adecuados y el programa empieza de nuevo.

El programa es sencillo de copiar, pero hay que poner atención con los numerosos caracteres gráficos y espacios en blanco entre ellos, que pueden dar lugar a errores.

VIC-20

```
0 REM***TOPOS***
3 SC=0
5 CC=30720:X=0:S4=36877:S1=36874:DB=0:S3=36876:S=234
8 POKE36878,15
26 POKE36879,187:PRINT"▣"
28 PRINT"▣▣▣ ▣   ■ ◤ ▀ ▣ ▀ ◤ ▀ ◤ ▀"
30 PRINT"  ▣ ■  ▣ ■ ▣ ■ ▣ ■ ▣ ■ ▣ ■ ▣ ■ ▣ ■ ◥"
40 PRINT"  ▣ ■  ▣ ■ ▣ ■ ▣  ◤ ▣ ■ ▣ ■ ◥▣ ▀"
50 PRINT"  ▣ ■  ▣ ■ ▣ ■ ▣ ■   ▣ ■ ▣ ■ ▣▀ ▣ ■"
60 PRINT"  ▣ ■  ◥▣ ◤ ▣ ■  ◥▣ ◤ ◥▣ ◤●"
70 PRINT"▣▣       ▣◤       ▀"
80 PRINT"       ◤          ■"
90 PRINT"      ◤O          ■"
100 PRINT"       ● ▣           ◤"
110 PRINT"          ◥▣▀   ◥▣▀"
115 PRINT"▣"
120 PRINT"▣  TIENES 15 PIEDRAS"
130 PRINT"▣ PARA APLASTAR TOPOS"
140 PRINT"▣▣PULSA G PARA JUGAR"
150 GETG$:IFG$<>"G"THEN150
160 POKE36879,255:PRINT"▣";
170 PRINT"▣▣▣▣▣▣▣▣▣▣▣▣▣▣▣▣▣▣▣▣▣▣   ▣▀▣▀  ▣▀▣▀  ▣▀▣▀ ";"▣"
172 TI$="000000"
```

```
175 B=7723:F=0
180 PRINT"S   ***F7-DISPARA***"
185 REM***BOMBAS***
190 IF PEEK(197)=63 THEN F=1: GOTO500
192 IF TI$>="000200"THEN PRINT"SQQQQ  ACABO EL TIEMPO!"
193 IFTI$>="000200"THEN POKE 36878,15:FORX=1TO999:POKES1,215:NEXTX:POKES1,0
194 IF TI$>="000200"THENPOKE36878,0:FORX=1TO5000:NEXTX:RUN
195 POKEB,32
200 IF B+1>7745 THEN POKEB,32:B=7723
210 B=B+1:POKEB,81:POKEB+CC,4
390 REM***MOVIMIENTO TOPOS***
400 MH=INT(RND(1)*3+1)
405 POKEX,121
410 ON MH GOTO420,440,460
420 POKE8169,65:POKE8169+CC,0:X=8169
425 FORT=1TO35:NEXTT
427 IFF=1 THEN GOTO500
430 GOTO 190
440 POKE8175,65:POKE8175+CC,0:X=8175
445 FORT=1TO35:NEXTT
447 IFF=1 THEN GOTO500
450 GOTO 190
460 POKE8181,65:POKE8181+CC,0:X=8181
465 FORT=1TO35:NEXTT
467 IFF=1 THEN GO TO 500
470 GOTO190
500 REM***CAIDA DE BOMBAS***
504 IFDB=15THENPRINT"SQQQQ   NO MAS BOMBAS!!":FORT=1TO5000:NEXT:RUN
505 IF PEEK(B+22)=65 THENGOTO600
510 IF PEEK(B+22)<>32 GOTO700
515 IFB+22>8164THEN GOTO700
520 POKEB,32
530 B=B+22:POKEB,81:POKEB+CC,4
532 IFS=200 THEN S=234:GOTO190
533 POKES3,S-1:FORT=1TO25:NEXTT:S=S-1
540 GOTO390
600 REM***APLASTA TOPOS***
601 POKES3,0
610 POKEB,32:POKEX,42:POKEX+CC,2
620 FORS=1TO100:POKES4,155:NEXTS:POKES4,0
623 POKEX,121:POKEX+CC,0
624 S=234
625 SC=SC+10
630 PRINT"SQ       SCORE=";SC
640 DB=DB+1:GOTO175
700 REM***EXPLOSION***
701 POKES3,0
710 POKEB,32:R=PEEK(B+22):U=PEEK(B+21):V=PEEK(B+23)
712 POKEB-2,46:POKEB,46:POKEB+2,46:POKEB-2+CC,5:POKEB+CC,5:POKEB+2+CC,5
715 POKEB+22,66:POKEB+21,77:POKEB+23,78:POKEB+22+CC,5:POKEB+21+CC,5:POKEB+23+CC,
5
720 FORS=1TO50:POKES1,175:NEXTS:POKES1,0
723 S=234
725 POKEB-2,32:POKEB,32:POKEB+2,32
727 POKEB+22,32:POKEB+21,32:POKEB+23,32
730 POKEB+22,R:POKEB+22+CC,0
732 POKEB+21,U:POKEB+21+CC,0
734 POKEB+23,V:POKEB+23+CC,0
740 DB=DB+1:GOTO175
```

Programas

Tranvía

VIC-20

Tranvía es un programa que pertenece a la familia de programas de tipo laberinto, y su nombre es este porque, el laberinto del que hay que salir, está inspirado en las intrincadas redes que forman las vías de los tranvías.

El laberinto que se genera al hacer funcionar el programa, es aleatorio, y será distinto cada vez. Una vez generado el laberinto, aparece en la línea superior de la pantalla un cuadradito, que es el tranvía, y el jugador tendrá que conducirlo hasta la línea inferior de la pantalla, utilizando las teclas que el programa indica en sus instrucciones. Para descender por el laberinto, se ha de escoger primero una línea de las que tienen su origen en la parte superior de la pantalla, utilizando las teclas G y H, y pulsando RETURN una vez que se ha escogido. Esta línea de origen no se puede abandonar hasta que se llega al final del laberinto o hasta que haya que renunciar (pulsando RETURN) por haber llegado a un callejón sin salida.

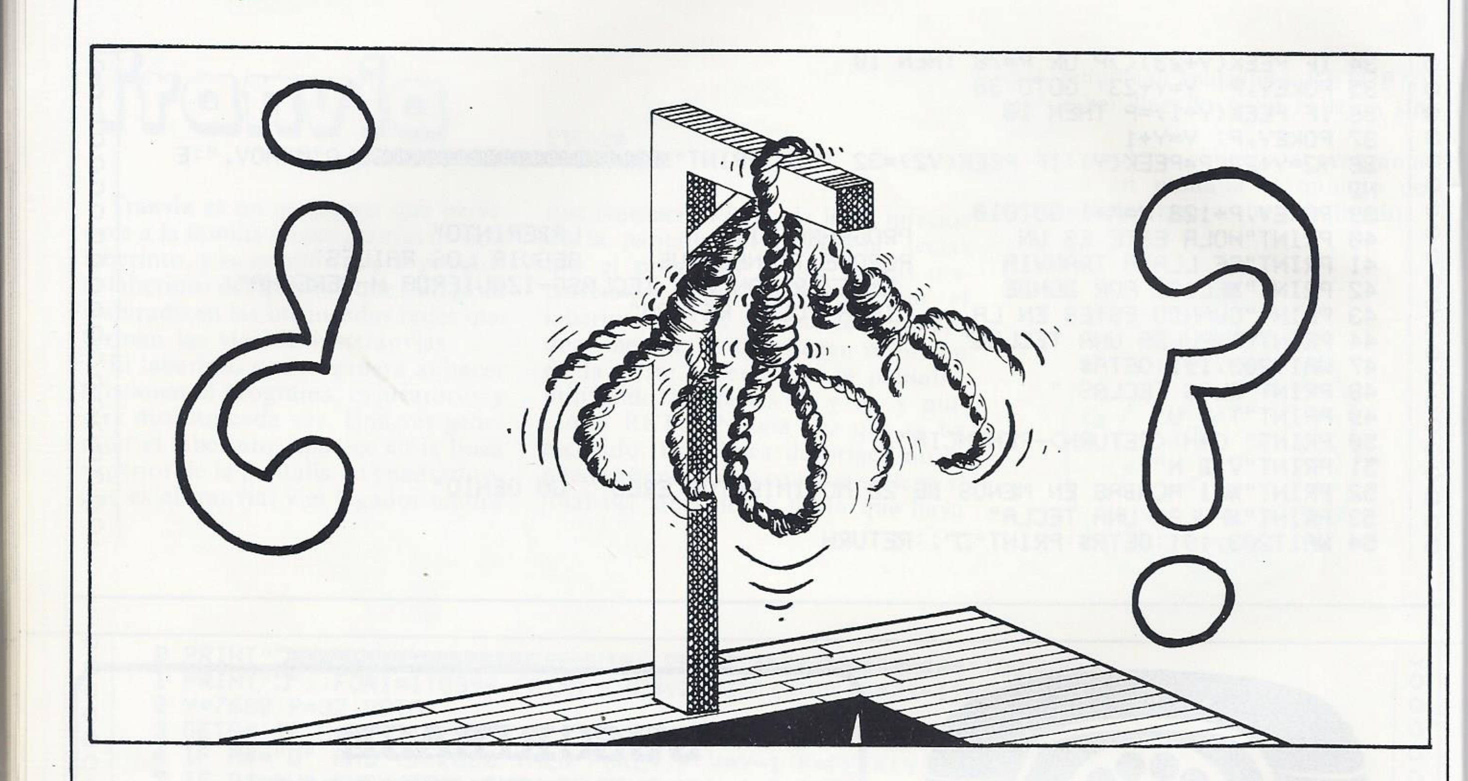
Al completar el juego, el programa muestra en pantalla el número de movimientos que se han realizado.




```
0 PRINT"TRANVIA":PRINT"#######":GOSUB40
1 PRINT"";:FORI=1TO396:PRINT CHR$(77+INT(RND(1)*2)+128);:NEXT:D=1
2 Y=7680:P=32:GOTO7
3 GETA$:IF A$="" THEN3
4 IF A$="G" AND Y> 7680 THEN POKEY,P:Y=Y-1:P=PEEK(Y)
5 IF A$="H" ANDY<7701 THEN POKEY,P:Y=Y+1:P=PEEK(Y)
6 IF A$=CHR$(13) THEN 10
7 POKE Y,P+128
8 GOTO3
9 REM
10 GET A$:IF A$="" THEN 10
11 IF A$="T" THEN D=1
12 IF A$="U" THEN D=3
13 IF A$="V" THEN D=5
14 IF A$="N" THEN D=7
15 IF A$="Y"THEN D=2
16 IF A$="B" THEN D=4
17 IF A$="G" THEN D=6
18 IF A$="H" THEN D=8
19 IF A$=CHR$(13) THEN PRINT"HAHAHAHAHAHA!":END
20 OND GOTO 22,24,26,28,30,32,34,36
21 GOTO10
22 IF PEEK(Y-23)<> PORP=78 THEN 10
23 POKEY,P:Y=Y-23:GOTO38
24 IF PEEK(Y-22)=P THEN 10
25 POKEY,P:Y=Y-22:GOTO38
26 IF PEEK(Y-21)<>P OR P=77 THEN 10
27 POKEY,P:Y=Y-21:GOTO38
28 IF PEEK(Y+22)=P THEN 10
29 POKEY,P:Y=Y+22:GOTO38
30 IF PEEK(Y+21)<>P OR P=77 THEN 10
31 POKEY,P:Y=Y+21:GOTO38
32 IF PEEK(Y-1)=P THEN 10
33 POKEY,P:Y=Y-1:GOTO 38
```

```
34 IF PEEK(Y+23)<>P OR P=78 THEN 10
35 POKEY,P: Y=Y+23: GOTO 38
36 IF PEEK(Y+1)=P THEN 10
37 POKEY,P: Y=Y+1
38 Y2=Y+22:P=PEEK(Y):IF PEEK(Y2)=32 THEN PRINT"▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒SOLO"M"MOV.":E
ND
39 POKEY,P+128:M=M+1:GOTO10
40 PRINT"HOLA ESTE ES UN        PROGRAMA TIPO          LABERINTO"
41 PRINT"SE LLAMA TRANVIA       PORQUE TIENES QUE      SEGUIR LOS RAILES"
42 PRINT"▒ELIGE POR DONDE        EMPEZAR CON LAS TECLASG-IZQUIERDA H-DERECHA"
43 PRINT"CUANDO ESTES EN LA     LINEA PULSA RETURN"
44 PRINT"▒▒PULSA UNA TECLA▒"
47 WAIT203,191:GETA$
48 PRINT"▒LAS TECLAS:"
49 PRINT"T Y U"
50 PRINT" G*H <RETURN>-RENUNCIA";
51 PRINT"V B N"
52 PRINT"▒SI ACABAS EN MENOS DE 20 MOVIMIENTOS ERES   UN GENIO"
53 PRINT"▒PULSA UNA TECLA"
54 WAIT203,191:GETA$:PRINT"▒";:RETURN
```

Concurso



El ahorcado

VIC-20

Jesús Fernández, lector de nuestra revista en Salamanca, nos envía este juego del ahorcado. Este conocido pasatiempo consiste en adivinar una palabra oculta, pensada por otro jugador. Para adivinarla sólo se dispone de una serie de intentos, en los que hay que ir ensayando letras, una por una, hasta acercar con todas las que contiene la palabra. Al comenzar el programa, uno de los jugadores introduce la palabra o frase que el otro jugador debe adivinar. Esta frase, no aparece en la pantalla al introducirla, y queda oculta hasta que sea adivinada, o hasta que se acaben los intentos. El programa incluye diversos sonidos musicales según que se acierte o no con las letras ensayadas. Al terminar el juego se indica el número de intentos realizados.

```
1 REM**JUEGO DEL AHORCADO**JESUS FERNANDEZ
5 POKE36879,110
10 PRINT"■■■  JUEGO DEL AHORCADO  "
15 GOSUB350
20 PRINT"■■■■■ESCRIBE UNA PALABRA"
30 GETA$:IFA$=""THEN30
40 IFA$=CHR$(13)ANDLEN(P$)>0THEN60
41 IFA$=" "THEN45
43 IFA$<"A"ORA$>"Z"THEN30
45 P$=P$+A$
```

```
50 GOTO30
60 B=LEN(P$)
61 PRINT"▒▒▒▒▌▌▚PISTA■                  ";:GOTO66
62 GETA$:IFA$=""THEN62
63 IFA$=CHR$(13)THEN70
64 PRINTA$;:GOTO62
66 PRINT"▌▌▌▌▌▌▌▌▌▌▌▌▌▌";:GOTO62
70 FORI=1TOB
80 H$=MID$(P$,I,1)
85 IFH$=" "THENH$=" ":GOTO100
90 H$="*"
100 PRINT"▒▒▒▒▒▒▒▒";TAB(I+1);H$:N$=N$+H$
110 NEXTI
120 PRINT"▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▌▌▚LETRA■";
125 A$="":GETA$:IFA$=""THEN125
126 IFA$<"A"ORA$>"Z"THEN125
128 IFA$=CHR$(13)THEN125
129 GOSUB700
130 PRINT"▌";A$:GOSUB740:F=F+1
135 GOSUB200
136 S=0
140 FORI=1TOB
150 IFA$=MID$(P$,I,1)THENGOSUB400:PRINT"▒▒▒▒▒▒▒▒";TAB(I+1);A$:GOSUB230:S=1
170 IFN$=P$THEN300
180 NEXTI
183 IFS<1THENGOSUB500:C=C+1:ONCGOTO 800,810,810,810,810
185 IFC=5THEN250
190 GOTO120
200 FORL=1TO1000:NEXTL
210 PRINT"▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒                         "
220 RETURN
230 N$=(LEFT$(N$,I-1)+A$+RIGHT$(N$,B-I)):RETURN
250 GOSUB600
255 PRINT"▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▚PERDISTE HAS TENIDO■ ▒    "C " ▚FALLOS■";
260 FORL=1TO2000:NEXTL
265 PRINT:PRINT"▚▚LA PALABRA ERA■        ▒                    ":PRINT P$
267 IFF>9THENPRINT"▚"
270 PRINT"▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒OTRA PARTIDA? (S/N) "
280 GETA$:IFA$=""THEN280
285 IFA$="S"THENRUN
290 IFA$<>"N"THEN280
295 PRINT"▚":END
300 PRINT"▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▌▌▚ENHORABUENA■"
303 GOSUB600
305 PRINT"▒▚UTILIZASTE■"F"▚INTENTO■";
306 IFF<>1THENPRINT"▒S■"
310 FORM=1TO2000:NEXT:GOTO267
350 FORJ=1TO22
360 POKE7987+J,119:POKE38705+J,1
370 NEXTJ
380 RETURN
400 REM**MUSICA ACIERTO**
405 PRINT"▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▌▌▚ACIERTO■";
410 POKE36878,15
420 FORL=1TO10
425 FORJ=1TO20
430 POKE36875,200
435 NEXTJ
436 POKE36875,140
```

PREMIADO CON
5.000
PESETAS

Concurso

Viene de la página anterior

```
440 NEXTL
450 POKE36878,0
455 PRINT"▮■▮  "
456 POKE36875,0
460 RETURN
500 REM**MUSICA FALLO**
502 PRINT"■QQQQQQQQQQQQQQQQQQ▮▮▮▮■FALLO■"
505 POKE36878,15
510 FORA=100TO200STEP.7
520 L=INT(RND(1)*50)+A
530 POKE36876,L+2
540 NEXTA
550 POKE36878,0
555 POKE36876,0
590 RETURN
600 REM**MUSICA FINAL**
605 POKE36878,15
610 FORA=250TO200STEP-2
615 POKE36876,A+2
620 FORM=1TO100:NEXTM
625 NEXTA
630 FORA=200TO250STEP2
635 POKE36876,A
640 FORM=1TO100:NEXTM
645 NEXTA
650 POKE36878,0
680 POKE36876,0
690 RETURN
700 REM**BUCLE COMPARACION**
710 FORO=1TOLEN(W$)
720 IFA$=MID$(W$,O,1)THEN125
730 NEXTO
735 W$=W$+A$:RETURN
740 PRINT"■■QQQQQQQQQQQQQQQQQQQQQ";TAB(1+LEN(W$));A$:PRINT"■":RETURN
800 G=8119:X=8185:GOTO850
810 G=G-88:X=X-88 :GOTO850
850 FORV=GTOXSTEP 22
860 POKEV,102:NEXTV
870 GOTO185
```

Ataque marciano

VIC-20

Los marcianitos atacan de nuevo, esta vez a través del programa que recibimos de José M. Campos, desde Pontevedra. El juego consiste en desplazarse con una nave a través de un escenario de planetas y satélites, intentando destruir el mayor número posible de naves marcianas, teniendo buen cuidado de no chocar con ningunao de los planetas, satélites o asteroides que hay por los alrededores. El jugador dispone de tres vidas, o naves, que debe hacer durar tanto como sea posible. El programa se carga en dos partes, cada una de las cuales hay que teclear y guardar en el *cassette* por separado.

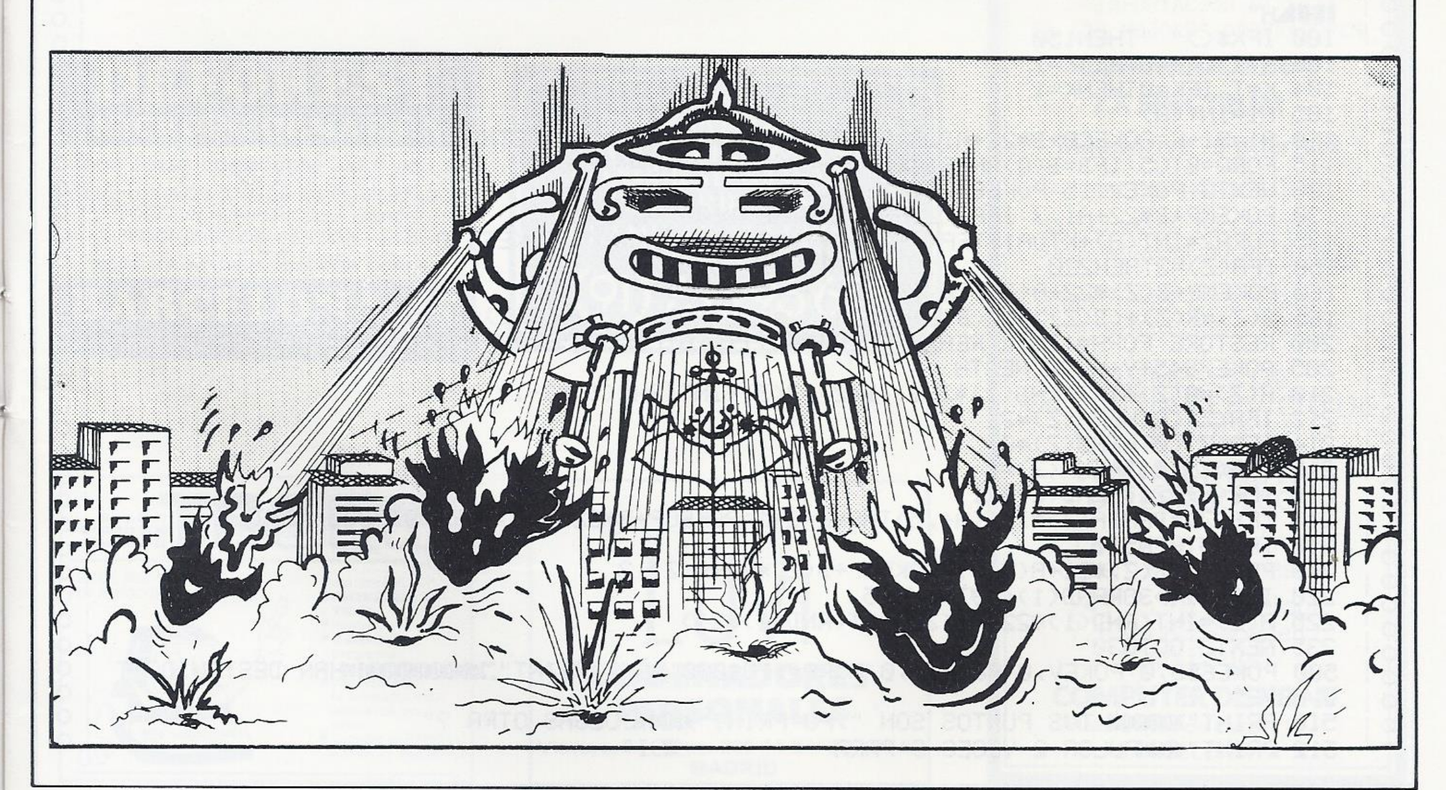
Para jugar se carga la primera parte, se escribe RUN con lo que aparece la presentación y las instrucciones y al terminar se nos pide que carguemos la segunda parte, que conviene guardar en la cinta inmediatamente a continuación de la primera.

En su versión original, uno de los planetas lleva un satélite que gira a su alrededor; esto conlleva una disminución de la velocidad del juego. José M., el autor del programa, sugiere que se elimine este efecto del satélite, anulando las líneas 1.000, 200 y 205 y modificando el final de la línea 160 a...: IF RND (1) <. 4 THEN 210.

Así mismo, y si se desea añadir un contador de tiempo al programa, se puede hacer añadiendo al final de la línea 17 .... : TI$ = "000000" e incluyendo las líneas siguientes:
99 PRINT" (HOME) (CTRL ON) (1 ESPACIO) (21 CRSR DOWN)" VAL(TI$) 211 IF TI$ > "000200" THEN 500

PREMIADO CON 5.000 PESETAS

```
5 POKE52,28:POKE56,28:CLR
10 FORX=7168TO7239:READD:POKEX,D:NEXT
20 FORI=7424TO7431:POKEI,0:NEXT
30 POKE36879,8:PRINT"":PRINT:PRINT:PRINT
40 PRINT" ";
50 PRINT" ";
60 PRINT" ":PRINT:PRINT
70 PRINT" ";
80 PRINT" ";
90 PRINT" "
95 PRINT"      POR JOSE CAMPOS":PRINT"      MAYO DE 1984":PRINT"PULSA UN
A TECLA"
100 POKE198,0:WAIT198,1
120 POKE36869,255:PRINT"  ATAQUE MARCIANO   ":PRINT:PRINT
130 PRINT" DEBERAS DESTRUIR EL   MAYOR NUMERO POSIBLE  DE NAVES ENEMIGAS"
140 PRINT" ESTA ES TU NAVE A":PRINT" Y ESTA B EL ENEMIGO."
145 PRINT" CUIDADO! NO CHOQUES   CON LOS PLANETAS DE FG   NI SUS SATELITE
S H"
150 PRINT" PARA MOVERTE USA":PRINT"     A,D,O,< Y SPACE":PRINT"PULSA UN
A TECLA
160 POKE198,0:WAIT198,1:POKE650,128:POKE36869,240:PRINT"LOAD"
170 POKE198,3:POKE632,19:POKE633,13
200 DATA4,26,99,130,130,99,26,4,32,88,198,65,65,198,88,32
210 DATA24,126,255,219,126,60,0,0,0,0,0,255,0,0,0,0
220 DATA3,15,31,59,127,127,239,175,192,240,248,252,62,246,255,239
230 DATA254,254,127,123,63,31,15,3,255,249,255,222,188,248,240,192
240 DATA0,0,24,60,44,24,0,0
```



Concurso

Ataque marciano II


```
5 CLR
10 POKE36879,8:PRINT"█▌       RECUERDA"
15 PRINT"█ PARA MOVERTE USA:":PRINT"██ A,D,O,< Y SPACE":PRINT"████PULSA UNA TEC
LA"
16 POKE198,0:WAIT198,1:POKE650,128:PRINT"▌":MP=7680:CP=38400:E=1:T=1:A=5:B=10:PU
=0
17 POKE36869,255:POKE36879,8:V=36878:S4=36877
18 DIMA(5),B(5):FORZ=0TO5:A(Z)=3:NEXT
30 FORZ=0TO5:GETX$:IFX$=""ANDT=0THENPOKE36877,0:GOTO96
40 POKEV,8:POKES4,240:POKEMP+B*22+A,32:A=A+T/E
50 T=T+(X$="A")-(X$="D"):IFT>1THENT=1
55 IFT<-1THENT=-1
57 B=B+(X$="O")-(X$=",")
60 IFA<0THENA=21
65 IFB<0THENB=0
70 IFA>21THENA=0
75 IFB>22THENB=22
80 POKECP+B*22+A,1:N=1:IFT=-1THENN=0
85 IFPEEK(MP+B*22+A)=32THEN95
90 E=E+1:FORQ=190TO140STEP-.5:POKES4,Q:NEXT:FORQ=0TO200:NEXT:POKES4,0
92 POKECP+B*22+A,2:FORQ=1TO300:NEXT:IFE>3THEN500
95 POKEMP+B*22+A,N
96 PRINT"███▌HI█DE█▌FG█H█████████████DE█▌FG███████████████████████HDE█F
G███EH"
97 PRINT"███████████DH█DE█▌FG█H████████████████DE█▌FG.TH████████H.TT█EH"
98 PRINT:PRINT:PRINT:PRINT:PRINT"███████H████H████████EH█████H████████H█████
███H"
100 IFX$<>" "THEN150
103 K=T:IFK=0THENK=1
104 K=T:IFK=0THENK=1
105 A1=INT(A)
110 A1=A1+K:POKECP+B*22+A1,7:P=PEEK(MP+B*22+A1):IFP=32THEN130
115 FORQ=0TO5:IFB=B(Q)ANDA1=A(Q)THENA(Q)=3:PU=PU+10
120 NEXTQ:POKEV,15:POKES4,130:GOTO135
130 POKEMP+B*22+A1,3:IFA1<21ANDA1>0THEN110
135 FORA2=INT(A)+KTOA1STEPK:POKEMP+B*22+A2,32:NEXT:POKES4,0
150 IFA(Z)=3THEN220
160 POKEMP+B(Z)*22+A(Z),32:IFRND(1)<.4THEN200
165 B(Z)=B(Z)+(B(Z)>B)-(B(Z)<Y):A(Z)=A(Z)+(A(Z)>A)-(A(Z)<A):GOTO217
200 RESTORE:FORH=1TO12:READY:POKE7845+Y+H,8:POKE38565+Y+H,3:FORV=0TO60:NEXTV
205 POKE7845+Y+H,32:NEXTH
210 A(Z)=A(Z)+INT(RND(1)*3-1):B(Z)=B(Z)+INT(RND(1)*3-1)
212 IFA(Z)<0THENA(Z)=21
214 IFB(Z)<0THENB(Z)=0
215 IFA(Z)>21THENA(Z)=0
216 IFB(Z)>21THENB(Z)=21
217 IFB(Z)=BANDA(Z)=INT(A+.5)THENE=E+1:POKECP+B*22+A,2:FORQ=0TO300:NEXT
218 IFE>3THEN500
219 POKECP+B(Z)*22+A(Z),1:POKEMP+B(Z)*22+A(Z),2
220 IFA(Z)<>3ORRND(1)<.9THEN235
225 A(Z)=INT(RND(1)*22):B(Z)=INT(RND(1)*21)
235 NEXTZ:GOTO30
500 POKE650,0:POKEV,0:POKES4,0:FORR=1TO1000:NEXT:PRINT"▌████████HAN DESTRUIDO T
U NAVE"
510 PRINT"█████ TUS PUNTOS SON ";PU:PRINT"████JUEGAS OTRA ?"
512 PRINT"███PULSA 2 VECES S PARA          ██SI"
```

```
530 GETX$:IFX$=""THEN530
535 IFX$="S"THENRUN16
540 IFX$="N"THENPOKE36869,240:END
550 GOTO530
1000 DATA1,1,23,45,66,86,106,104,80,56,33,11
```

Aclaración

En el numero 4 de Commodore Magazine, y por un lamentable error,en el programa titulado CARTAS aparecia, como texto, el comentario de otro programa titulado DIBUJO JOY. Rectificamos el error y adjuntamos el comentario original correspondiente al programa CARTAS.

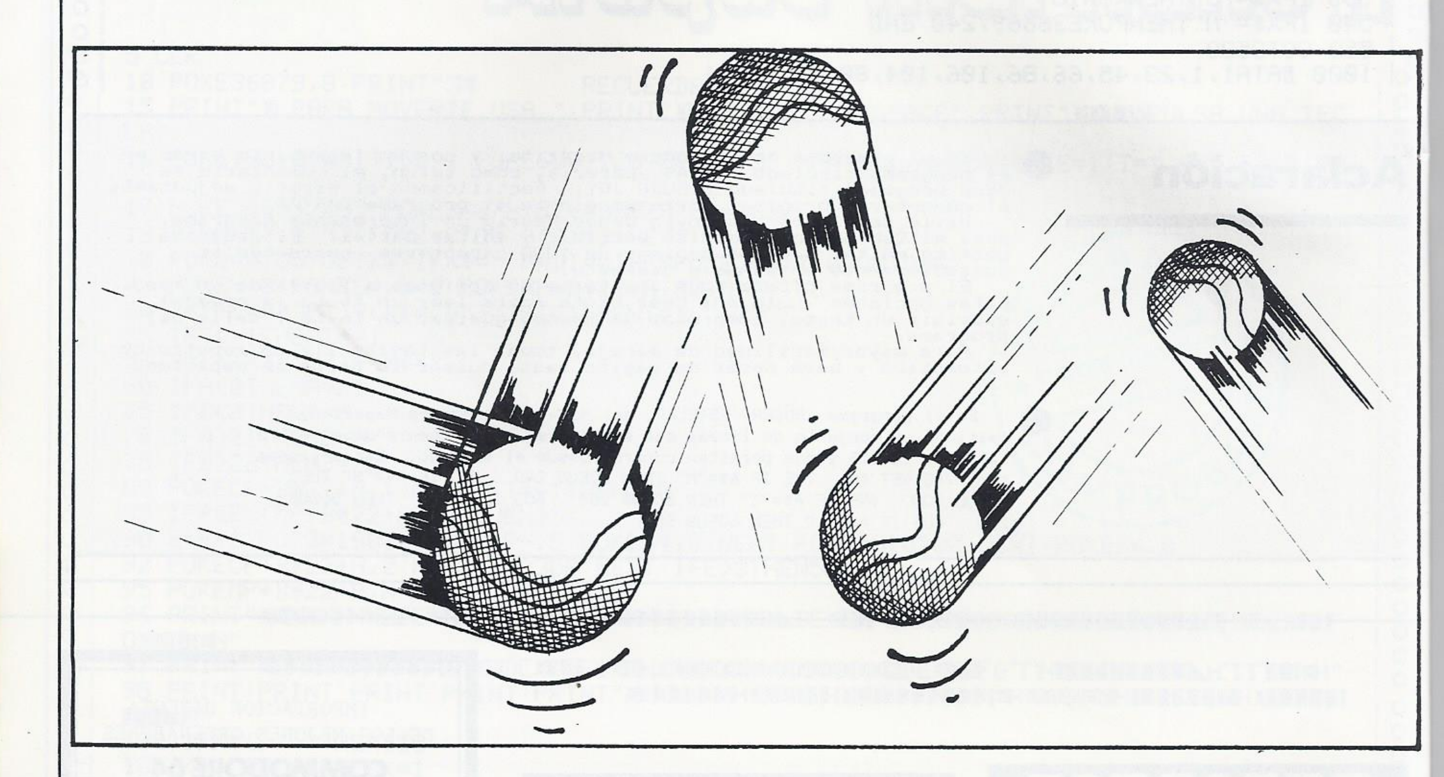
David Caballero nos envio desde Madrid un interesante programa, para el CBM 64, que permite escribir y editar cartas. El programa permite editar hasta 4 paginas de 1000 caracteres guardandolas posteriormente en cinta o diskette.

El programa ofrece toda una serie de opciones a traves de un menu. Estas opciones incluyen: Leer cinta (para leer un texto ya creado), escribir un texto, comprobar un texto, guardar un texto y salir del programa.

Para mayor facilidad de manejo, todas las teclas llevan repeticion automatica y para pasar de pagina basta pulsar la barra de espacios.

En el programa CRUCERO ESTELAR, del No.4 de Commodore Magazine, omitimos un conjunto de lineas que sustituyen a las comprendidas entre la 780 y la 805 y que permiten control desde el teclado. Estas lineas son: 780 GET A$ 785 IF A$="M" THEN GOSUB 240 790 IF A$="N" THEN GOSUB 200 795 IF A$="Z" THEN GOSUB 280 800 IF A$="A" THEN GOSUB 270 805 IF A$=" " THEN GOSUB 500

Concurso



Pelota loca

```
3 DIME(10)
5 POKE36878,14
10 FORX=0TO25:PRINT"▪                    ":NEXT
20 POKE36879,42
25 E(0)=16:E(1)=5:E(2)=12:E(3)=15:E(4)=20:E(5)=1:E(6)=32
26 E(7)=12:E(8)=15:E(9)=3:E(10)=1
27 FORDS=0TO10
28 POKE7883+DS,E(DS):FORF=0TO250:NEXTF:NEXTDS
30 FORX=0TO82STEP2
40 POKE7894+X,1:POKE36876,195:FORZX=0TO75:NEXTZX
50 POKE7894+X,32:POKE36876,0:NEXT
55 POKE7894+X,1
60 FORW=0TO7
65 POKE7894+X+38400-7680,W:FORHL=0TO200:NEXT
70 NEXT
80 POKE36879,27:IFM1>0THEN330
90 POKE36879,27:G=38400-7680
100 FORX=0TO25:PRINT"▪                    ":NEXT
110 FORS=7726TO7743:POKES,99:NEXT:POKE7725,79:POKE7744,80
120 FORS=7747TO8187STEP22:POKES,101:POKES+19,103:NEXT
130 FORS=7770TO7787:POKES,9:NEXT
140 FORS=7792TO7809:POKES,22:NEXT
```

VIC-20

PELOTA LOCA es un juego para el **VIC 20**, basado en el conocido juego de los "ladrillos", en el que se trata de derribar una pared de ladrillos mediante una pelota y una raqueta controlada por el jugador, y que nos envía José Constanti desde Tarragona. La raqueta se controla mediante las teclas C y Z que permiten desplazarla por la línea inferior de la pantalla. El juego presenta dos particularidades interesantes: la primera de ellas es que la pelota puede variar su trayectoria ascendente en cualquier momento y de forma aleatoria, lo que complica el juego. La segunda consiste en tres postes que se interponen entre la pelota y la pared a derribar, y que hacen más complicado llegar hasta la pared. Con todo esto pueden pasarse unas cuantas horas entretenidas hasta acabar con todos los ladrillos.

```
150 FORS=7814TO7831:POKES,9:NEXT:POKE7823,22:POKE7822,22
160 FORS=7836TO7853:POKES,9:NEXT:POKE7845,3:POKE7844,3
165 POKE7839,22:POKE7840,22:POKE7849,22:POKE7850,22
170 POKE7866,85:POKE7867,73:POKE7888,74:POKE7889,75
180 POKE7949,85:POKE7950,73:POKE7959,85:POKE7960,73
185 POKE7971,74:POKE7972,75:POKE7981,74:POKE7982,75:M1=0
190 PRINT"SSS  PUNTUACION =";M1
200 A=8146:P=7937:D=21:POKE36878,15:M1=0
205 G=38400-7680
210 POKEA-1,70:POKEA,70:POKEA+1,70
215 POKEA-1+G,4:POKEA+G,4:POKEA+1+G,4
220 POKEP,81
225 POKEP+G,2
230 IFPEEK(P+D)<>32THEN 1000
240 POKEP,32:P=P+D:POKEP,81:POKEP+G,7
250 IFPEEK(197)=33ORPEEK(197)=34THEN 252
251 GO TO 260
252 GO TO2000
260 PRINT"SSS  PUNTUACION= ";M1
275 RS=RND(1)
280 IFRS<0.9THEN290
283 IFD=-22THEND=-21:GOTO287
284 IFD=-21THEND=-23:GOTO287
285 IFD=-23THEND=-22:GOTO287
286 GOTO290
287 POKE36876,195:FORCX=0TO15:NEXT:POKE36876,0
290 FORRR=0TO100:NEXT
300 IFP<8147THEN230
319 M1=M1+1
320 GOTO5
330 IFM1<151THEN390
340 FORS=0TO25:PRINT"                          ":NEXT
350 PRINT"SQQQQQQQQQQ]]]BUENA PUNTUACION"
353 PRINT"]]]]]]]]]]]]SIGUE";M1-1
354 FORLL=148TO220STEP.7:POKE36876,LL:NEXT
355 FORLL=128TO200:POKE36876,LL:NEXT
356 FORLL=200TO128STEP-1:POKE36876,LL:NEXT:POKE36876,0
357 FORJJ=0TO2000:NEXT
360 GOTO90
390 PRINT"S                    "
395 POKE36874,170:POKE36877,240
396 PRINT"SQQQQQQQQQQQQ]]]]]LASTIMA ";M1-1
397 FORLL=0TO5000:NEXT:POKE36874,0:POKE36877,0:POKE36878,0:END
415 PRINT"                    "
1000 IFPEEK(P+D)<>101THEN1020
1010 IFD=-23THEND=-21:GOTO1020
1015 D=23
1020 IFPEEK(P+D)<>103THEN1040
1030 IFD=-21THEND=-23:GOTO1040
1035 D=21
1040 IFPEEK(P+D)=73THEND=-21
1050 IFPEEK(P+D)=74THEND=21
1060 IFPEEK(P+D)=75THEND=23
1070 IFPEEK(P+D)=85THEND=-23
1080 IFPEEK(P+D)=79THEND=23
1090 IFPEEK(P+D)=80THEND=21
1100 IFPEEK(P+D)<>99THEN1120
1110 IFD=-21THEND=23:GOTO1120
1111 IFD=-22THEND=22:GOTO1120
```

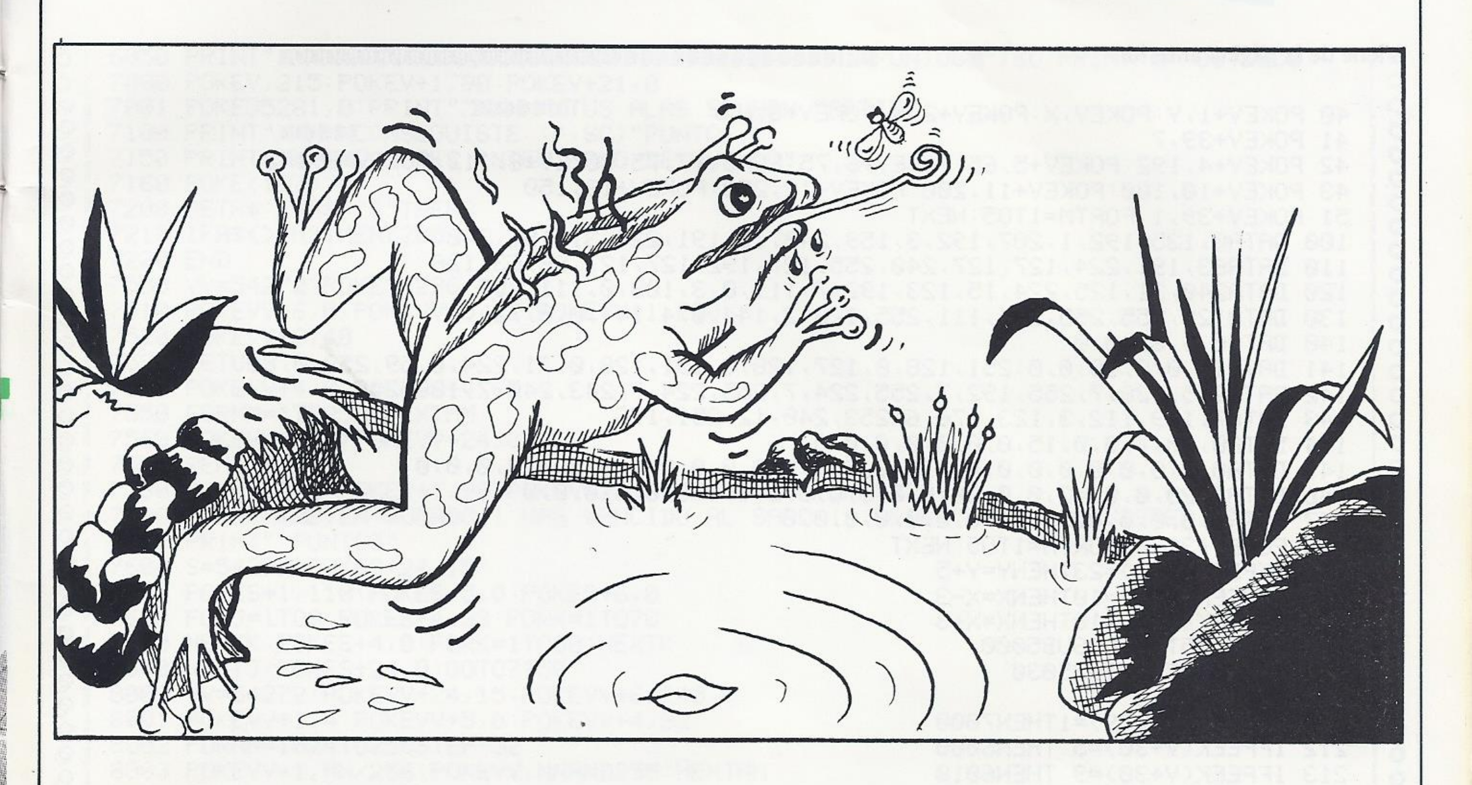
PREMIADO CON 5.000 PESETAS

Programas

Viene de la página anterior

```
1115 D=21
1120 IFPEEK(P+D)<>70THEN1200
1140 IFP+D=ATHEND=-22:GOTO1151
1050 IFPEEK(P+D)=74THEND=21
1060 IFPEEK(P+D)=75THEND=23
1070 IFPEEK(P+D)=85THEND=-23
1080 IFPEEK(P+D)=79THEND=23
1090 IFPEEK(P+D)=80THEND=21
1100 IFPEEK(P+D)<>99THEN1120
1110 IFD=-21THEND=23:GOTO1120
1111 IFD=-22THEND=22:GOTO1120
1115 D=21
1120 IFPEEK(P+D)<>70THEN1200
1140 IFP+D=ATHEND=-22:GOTO1151
1145 IFP+D=A-1THEND=-23:GOTO1151
1150 D=-21:GO TO1151
1151 GOTO1000
1200 IFPEEK(P+D)<>9THEN1300
1210 POKEP+D,32
1215 IFD=-21THEND=23:GOTO1240
1220 IFD=-23THEND=21:GOTO1240
1225 IFD=21THEND=-23:GOTO1240
1230 IFD=23THEND=-21:GOTO1240
1234 IFD=22THEND=-22:GOTO1240
1237 IFD=-22THEND=22
1240 M1=M1+1
1260 GOTO1000
1300 IFPEEK(P+D)<>22THEN1400
1310 POKEP+D,32
1315 IFD=-21THEND=23:GOTO1340
1320 IFD=-23THEND=21:GOTO1340
1325 IFD=21THEND=-23:GOTO1340
1330 IFD=23THEND=-21:GOTO1340
1334 IFD=22THEND=-22:GOTO1360
1337 IFD=-22THEND=22
1340 M1=M1+5
1360 GOTO1000
1400 IFPEEK(P+D)<>3THEN1500
1410 POKEP+D,32
1415 IFD=-21THEND=23:GOTO1440
1420 IFD=-23THEND=21:GOTO1440
1425 IFD=21THEND=-23:GOTO1440
1430 IFD=23THEND=-21:GOTO1440
1434 IFD=22THEND=-22:GOTO1440
1437 IFD=-22THEND=22
1440 M1=M1+100
1460 GOTO1000
1500 PRINT"   PUNTUACION= ";M1
1515 POKE36876,215:FORS=0TO15:NEXT:POKE36876,0
1520 GOTO230
2000 IFPEEK(197)=33ANDA=8145THEN2060
2010 IFPEEK(197)=34ANDA=8160THEN2060
2020 POKEA-1,32:POKEA,32:POKEA+1,32
2030 IFPEEK(197)=33THENA=A-1
2040 IFPEEK(197)=34THENA=A+1
2050 POKEA-1,70:POKEA,70:POKEA+1,70
2055 POKEA-1+G,4:POKEA+G,4:POKEA+1+G,4
2060 GOTO260
```





El sapo venenoso

Commodore 64

En este juego el *C64* trata de evitar con un sapo que lanza "escupitajos mortales, que tú, controlando el movimiento de una mariposa, puedas recoger los **5** granitos de polen distribuidos por la pantalla.

El sapo escupe unos puntitos negros que te serán difíciles de apreciar en tu televisión. Además, hay un junco contra el que no puedes chocar o te destruirás.

Por supuesto, te ocurrirá lo mismo si te estrellas con los límites del recuadro dentro del que se puede mover tu mariposa. Las instrucciones están dentro del programa y no necesitas *Joystick* para jugar con él. Y... ¡cuidado con el sapo!

```
0 REM****************************
1 REM*** EL SAPO VENENOSO ***
2 GOSUB 9000
3 POKE53278,0:POKE53279,0:POKE53264,0
4 PRINT"]":SC=0:CLR
5 POKE53280,9:POKE53281,9:GOSUB500
10 V=53248:POKEV+21,255:POKE2040,13:POKE2041,14
21 POKE2042,15:POKE2043,15:POKE2044,15:POKE2045,15:POKE2046,15:POKE2047,15
25 POKEV+39,10:POKEV+40,5:POKEV+41,7:POKEV+42,7:POKEV+43,7:POKEV+44,7:POKEV+45,7
:POKEV+46,7
30 FORI=0TO190:READA:POKE832+I,A:NEXT
35 X=215:Y=90:M=254:N=200
```

Viene de la página anterior

```
40 POKEV+1,Y:POKEV,X:POKEV+2,M:POKEV+3,N
41 POKEV+39,7
42 POKEV+4,192:POKEV+5,65:POKEV+6,75:POKEV+7,75:POKEV+8,112:POKEV+9,142
43 POKEV+10,100:POKEV+11,200:POKEV+12,250:POKEV+13,150
51 POKEV+39,1:FORTM=1TO5:NEXT
100 DATA0,135,192,1,207,192,3,159,240,15,191,224,31,191,240
110 DATA63,191,224,127,127,240,255,127,192,127,127,96,63,126
120 DATA240,31,125,224,15,123,192,7,119,0,3,108,0,111,255
130 DATA128,255,255,224,111,255,128,2,144,0,4,144,0,8,72
140 DATA0,4,68,0,0
141 DATA14,0,0,10,0,0,251,128,0,127,128,0,191,128,0,71,224,0,59,252,0,31
142 DATA255,128,7,255,192,7,255,224,7,255,224,7,243,240,7,180,240
143 DATA7,119,112,3,123,176,6,253,240,12,251,176
144 DATA0,20,7,0,0,15,0,0,0,0,0,0,0
145 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
146 DATA0,20,0,0,42,0,0,20,0,0,8,0,0,0,0,0,0,0,0,0,0
147 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0
148 POKEV+39,0:FORTM=1TO5:NEXT
150 IFPEEK(197)=23THENY=Y+5
170 IFPEEK(197)=10THENX=X-3
180 IFPEEK(197)=18THENX=X+3
190 IFX>255THENGOSUB5000
191 IFX<0THENGOSUB5030
210 POKEV+39,4
211 IFPEEK(53279)=1THEN7000
212 IFPEEK(V+30)=5 THEN6000
213 IFPEEK(V+30)=9 THEN6010
214 IFPEEK(V+30)=17THEN6020
215 IFPEEK(V+30)=33THEN6030
216 IFPEEK(V+30)=65THEN6040
217 TT=INT(RND(0)*12)+1:IFTT=6THENGOSUB8000
218 POKEV+39,7
249 POKE53276,PEEK(53276)AND(255-2↑0)
250 Y=Y-2:GOTO40
500 FORU=1024TO1063:POKEU,94:NEXT
510 FORU=1103TO2023STEP40:POKEU,94:NEXT
520 FORU=2022TO1984STEP-1:POKEU,94:NEXT
530 FORU=1984TO1064STEP-40:POKEU,94:NEXT
540 FORP=55296TO55335:POKEP,10:NEXT
550 FORP=55375TO56295STEP40:POKEP,10:NEXT
560 FORP=56294TO56256STEP-1:POKEP,10:NEXT
570 FORP=56256TO55295STEP-40:POKEP,10:NEXT
575 FOR BR=1552TO1825STEP39:POKEBR,78:NEXT:POKE1357,78
576 FORBR=1396TO1513STEP39:POKEBR,224:NEXT
577 FORBR=55824TO56097STEP39:POKEBR,5:NEXT:POKE55629,5
578 FORBR=55668TO55785STEP39:POKEBR,12:NEXT
600 PRINT"[illegible]=PUNTOS[illegible]";SC:PRINT"[illegible]":RETURN
5000 X=X-255:POKEV+16,1:RETURN
5030 IFX<0THENX=X+255:POKEV+16,0:RETURN
6000 POKE53269,PEEK(53269)AND(255-2↑2):POKE53278,0:SC=SC+5:GOSUB7500:GOTO6049
6010 POKE53269,PEEK(53269)AND(255-2↑3):SC=SC+10:GOSUB7500:GOTO6049
6020 POKE53269,PEEK(53269)AND(255-2↑4):SC=SC+15:GOSUB7500:GOTO6049
6030 POKE53269,PEEK(53269)AND(255-2↑5):SC=SC+20:GOSUB7500:GOTO6049
6040 POKE53269,PEEK(53269)AND(255-2↑6):SC=SC+10:GOSUB7500:GOTO6049
6049 IFSC=60THEN7560
```

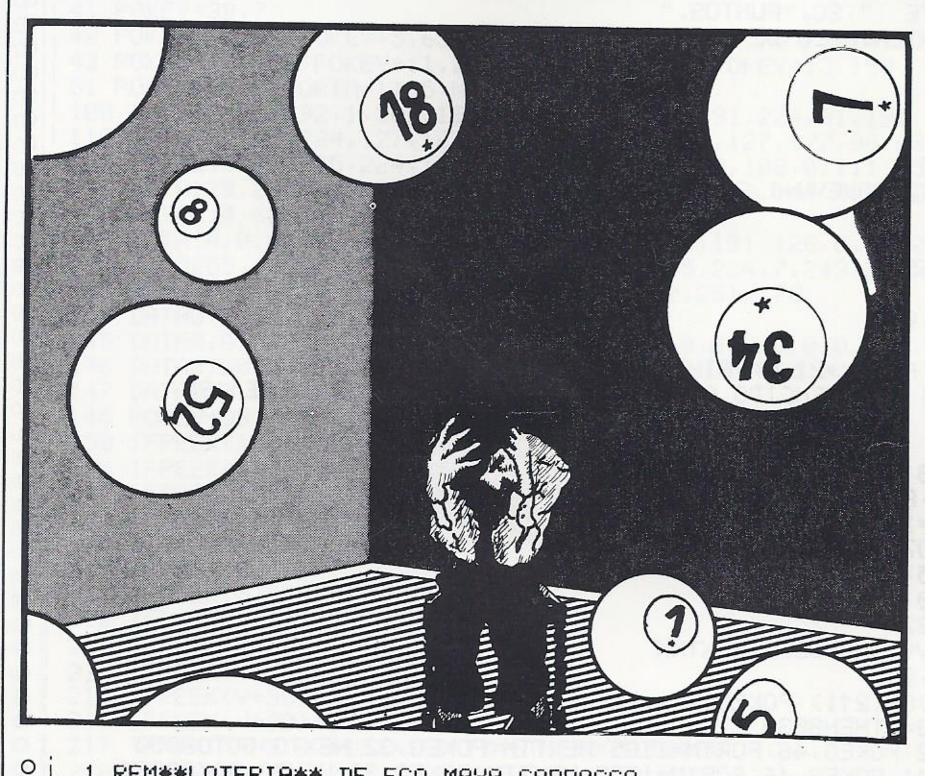
```
6050 PRINT"QQQQQQQQQQQQQQQQQQQQQQ]]]]]]]]]]]]]]]]]]]]]]]]]]]RPUNTOS■";SC:PRINT"■":GOTO250
7000 POKEV,215:POKEV+1,90:POKEV+21,0
7001 POKE53281,0:PRINT"♥QQQ]]]]TUS ALAS SE HAN PARTIDO......"
7100 PRINT"QQQ]]]CONSEGUISTE :";SC;"PUNTOS."
7150 PRINT"QQQ]]]QUIERES INTENTARLO DE NUEVO?"
7160 POKE(197),0
7200 GETA$:IFA$="S"THEN3
7210 IFA$<>"N"THEN7200
7220 END
7500 VV=54272:POKE54296,15
7510 POKEVV+6,0:POKEVV+5,12:POKEVV+1,200
7520 POKEVV+3,40
7530 RETURN
7540 POKEVV+4,0:POKEVV+4,65
7550 FORMM=1TO250:NEXTMM
7560 POKEVV+4,0:POKEVV+24,0
7570 REM
7580 POKEV,215:POKEV+1,90:POKEV+21,0:PRINT"♥":POKE53281,0
7590 PRINT"■QBIEN JUGADO!! HAS VENCIDO AL SAPO!!":PRINT"QQ]CONSEGUISTE";SC;
7591 PRINT" PUNTOS"
7600 S=54272:POKES+24,15
7610 POKES+1,110:POKES+5,0:POKES+6,0
7620 FORJ=1TO8:POKES+4,33:FORK=1TO70
7630 NEXTK:POKES+4,0:FORK=1TO80:NEXTK
7640 NEXTJ:POKES+24,0:GOTO7150
8000 VV=54272:POKEVV+24,15:POKEVV+6,240
8001 POKEVV+1,4:POKEVV+5,0:POKEVV+4,33
8002 FORNN=1024TO250STEP-32
8003 POKEVV+1,NN/256:POKEVV,NNAND255:NEXTNN
8004 POKEVV+4,0:POKE54296,0
8010 POKE53277,PEEK(53277)OR(2↑1):POKE53271,PEEK(53271)OR(2↑1)
8015 G=INT(2*RND(1))+1:IFG=2THEN8030
8020 FORG=1771TO1748STEP-2:POKEG,46:FORTM=1TO5:NEXTTM:POKEG,32:NEXTG:GOTO8050
8030 FORG=1771TO1074STEP-41:POKEG,46:FORTM=1TO5:NEXTTM:POKEG,32:NEXTG:GOTO8050
8050 POKE53277,PEEK(53277)AND(255-2↑1):POKE53271,PEEK(53271)AND(255-2↑1):RETURN
9000 PRINT"♥":POKE53281,0:PRINT"■QQ]]]]]]]]]]]]]]REL SAPO VENENOSO■"
9010 PRINT"QQQ]]]DEBERAS RECOGER 5 BOLAS DE POLEN"
9015 PRINT"QQ]QUE ESTAN DISTRIBUIDAS EN LA PANTALLA."
9020 PRINT"QQ]]RX■ IMPIDE QUE EL VIENTO TE SUBA"
9021 PRINT"]]]RA■ =IZQUIERDA"
9022 PRINT"]]]RD■ =DERECHA"
9030 PRINT"QQ]]TUS ALAS SON FRAGILES,NO TOQUES           EL BORDE NI EL JUNCO!."
9040 PRINT"QQQ]]CUIDADO CON EL SAPO!"
9050 PRINT"QQQQ]]]PULSA CUALQUIER TECLA PARA EMPEZAR"
9060 GETA$:IFA$=""THEN9060
9070 RETURN
```

PROGRAMA COMPROBADO commodore Magazine



Concurso

Loteria



Commodore 64

Loteria es el título de este programa, que sirve para jugar al bingo con el **CBM 64.** Nos ha sido enviado por **Francisco Moya Carrasco,** uno de nuestros lectores en Sevilla.

El funcionamiento del programa es el siguiente:

Al escribir RUN, después de copiado el programa, comienza la presentación del juego. Para comenzar se pulsa una tecla, apareciendo el primer número en pantalla, aleatorio entre 1 y 90, debiendo pulsarse asimismo una tecla para que vayan apareciendo los números sucesivos. Todos los números tienen un formato de 8 líneas × 10 columnas. Cuando se canta un bingo o una línea, hay 2 formas de pasar a comprobar lo cantado. La primera es pulsando la tecla "t", con lo que se obtiene una tabla con todos los números y entre ellos y en inverso, los que han aparecido.

Por otro lado, si la tecla pulsada es la "v", entonces podremos introducir uno a uno los números a comprobar, encargándose el programa de decirnos si han aparecido ya, o todavía no.



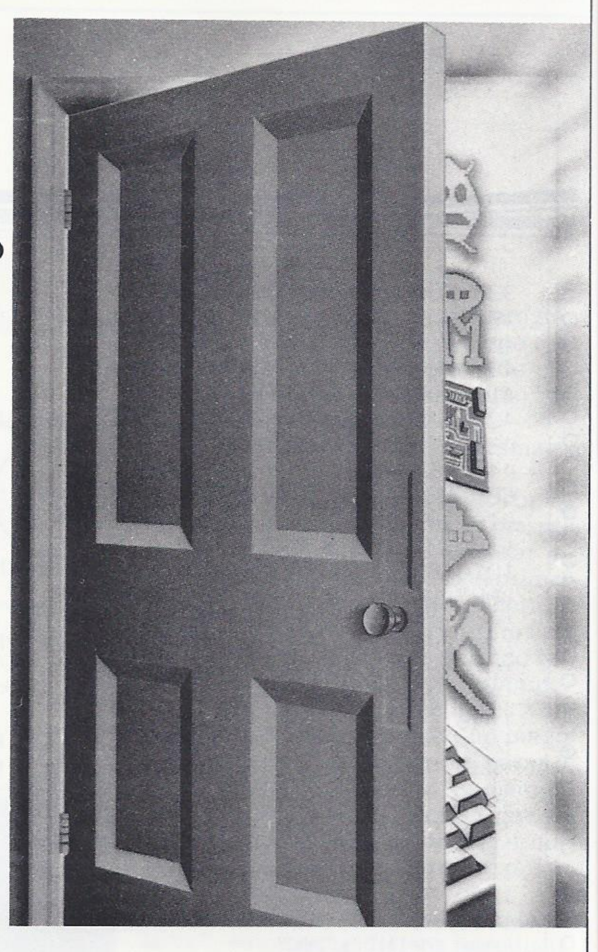

```
1 REM**LOTERIA** DE FCO MAYA CARRASCO
2 V=54270:POKEV+26,15:POKE53281,0:PRINT"▮":POKE53280,0
3 PRINT"♫":GOSUB3000
4 A$(1)=" ◪ ■ ":A$(2)="◪      ":A$(3)="◪■  ◪ ":A$(4)="◪■  "
5 DIMA(90)
7 A$(5)="    ◪ ":A$(6)="  ◪■ ":A$(7)=" ◪    ":A$(8)="◪    ■ ":A$(9)=" ◪■ "
10 B=0
15 B=B+1
20 C=INT(RND(0)*90)+1
30 A(B)=C
40 IFB=1THEN49
42 FORX=B-1TO0STEP-1
46 IFA(X)=CTHEN20
48 NEXTX
49 GOTO1200
50 GOSUB320
55 PRINT"▮▮▮▮▮▮▮▮▮▮▮▮▮▮PULSE UNA TECLA",,"▮▮▮▮▮▮▮▮▮▮▮LA V Y LA T PARA COMPROBAR"
57 FORW=1TO10:GETA$:NEXTW
60 GETA$:IFA$=""THEN60
63 PRINT"♫"
65 IFA$="V"THENGOSUB1010
67 IFA$="T"THENGOSUB3500
70 IFB<90THEN15
320 D=INT(C/10):E=C-D*10
325 PRINT"▮":IFD=0THEN360
```

PROGRAMA commodore Magazine COMPROBADO

```
330 FORR=1TOD*8:READZ:NEXT
360 FORP=1TO8:READZ:PRINTTAB(14)A$(Z):NEXTP
400 RESTORE
405 PRINT"■":IFE=0THEN420
410 FORR=1TOE*8:READZ:NEXT
420 FORP=1TO8:READZ:PRINTTAB(20)A$(Z):NEXTP
430 RESTORE
440 RETURN
600 DATA1,3,3,3,3,3,3,1,6,1,6,6,6,6,6,7,8,5,5,1,4,4,4,7
610 DATA1,3,5,1,5,5,3,1,4,4,4,3,2,5,5,5,7,4,4,1,5,5,5,8
620 DATA1,4,4,4,2,3,3,1,8,5,5,6,9,9,9,9,1,3,3,1,3,3,3,1,1,3,3,7,5,5,5,5
1010 PRINT"■■■■■■■COMPROBACION DE NUMEROS",,"■■■■POR FAVOR DEME NUMEROS A COMPRO
BAR"
1015 INPUT"NUMERO";AR
1020 FOR K=1 TO B:IF AR=A(K) THENPRINTTAB(20)"■■BIEN■":GOTO1300
1030 NEXT
1040 POKEV+6,129:POKEV+8,200:POKEV+3,2:POKEV+7,90
1046 FORH=1TO20
1048 PRINTTAB(19)"■■NO HA SALIDO":FORT=1TO150:NEXT
1049 PRINTTAB(19)"■■NO HA SALIDO":FORT=1TO150:NEXT:NEXT
1050 POKEV+6,128
1053 PRINT"■":PRINT"PULSE UNA TECLA PARA SEGUIR EL JUEGO" :GOTO60
1200 POKEV+6,17:POKEV+8,24:POKEV+7,84:POKEV+3,58:FORG=1TO120:NEXT:POKEV+6,16
1210 GOTO50
1300 POKEV+6,33:POKEV+8,100:POKEV+3,58:POKEV+7,58
1310 FORR=1TO100:NEXT:POKEV+6,34:GOTO1015
3000 PRINT"■■■■"
3010 PRINTTAB(13) "■          ■"
3013 PRINTTAB(13) "■ B I N G O ■"
3015 PRINTTAB(13) "■          ■"
3020 PRINTTAB(5)"■■DOS FORMAS PARA COMPROBAR BINGOS "
3025 PRINTTAB(5)"O LINEAS :"
3030 PRINTTAB(5)"■■V.- NUMEROS UNO A UNO■"
3032 PRINTTAB(5)"T.- TABLA DE NUMEROS APARECIDOS"
3034 PRINT"■■■■■■■■UNA TECLA PARA COMENZAR"
3036 GETA$:IFA$=""THEN3036
3045 PRINT"■■■■■■■■■TODOS LOS CARTONES YA RETIRADOS■■■■■"
3080 PRINT"MUCHA SUERTE";TAB(17)"■BINGO";TAB(28)"■MUCHA SUERTE■■■■■■■"
3081 PRINT"■■■■■LAS BOLAS SE ESTAN MOVIENDO■"
3082 FORT=1TO100:POKEV+6,129:POKEV+7,0:POKEV+8,0
3083 POKEV+3,INT(RND(8)*45):FORR=1TOINT(RND(1)*30):NEXT
3084 POKEV+6,128:FORJ=1TOINT(RND(9)*20):NEXT:NEXT
3095 PRINT"■■■■■■■UNA TECLA PARA EL PRIMER NUMERO"
3100 GETA$:IFA$=""THEN3100
3150 PRINT"■":RETURN
3500 PRINT"■ "
3510 FORP=1TO90
3515 FORT=1TOB
3520 IFP=A(T)THENPRINT" ■";A(T);"■■";:T=B:GOTO3525
3523 NEXTT
3524 IFP<>A(B)THENPRINTP;
3525 IFP<11THENPRINT" ";
3526 IFP-(INT(P/10))*10=0THENPRINT
3527 NEXTP
3530 PRINT"■■■■■TODOS LOS NUMEROS SOBRE BLANCO"
3531 PRINT"■■■■HAN SALIDO. PULSE UNA TECLA "
3532 PRINT"■■■■PARA CONTINUAR"
3533 GETA$:IFA$=""THEN3533
3534 PRINT"■":RETURN
```

PROGRAMA commodore Magazine COMPROBADO

Cómo diseñar juegos para ordenador (capitulo 4)



Desde la aparición del primer ordenador personal los gráficos han ido adquiriendo más y más importancia. Hoy en día uno de los factores para medir la calidad de un ordenador son los gráficos y lo mismo puede decirse de los programas (y sobre todo de los juegos), si tiene gráficos es bueno y si no es muy malo. ¿Es este punto de vista lógico?, la respuesta es ambigua, sí y no. Es válido si se considera que los gráficos proporcionan una información más clara y fácil de comprender, y dan más vistosidad al programa. Pero hay grandes juegos (La Aventura, también llamada La Caverna Colosal, por ejemplo) que no tienen gráficos, pero ofrecen más entretenimiento que muchos otros que tienen dibujos. En definitiva podemos decir que aunque no es fundamental, la existencia de gráficos mejora el juego.

Cuando un nuevo usuario del **Commodore 64** lee el manual que acompaña a la máquina observa, con desilusión, que no hace la menor referencia a gráficos (nos referimos naturalmente a gráficos en alta resolución y no a los cubitos que hay en las teclas). Sus posibilidades se reducen a combinaciones geométricas sencillas basadas en cubos, rectas y algún que otro cubo. Asimismo las posibilidades de color no son muy amplias, un borde blanco y un fondo azul, que parecen inmutables, crean en él (y mucho más cuando ve un juego comercial) el sentimiento de que las posibilidades de crear grandes dibujos se le escapan de las manos. ¿Es esto cierto? No, aunque el manejo de gráficas no sea tan sencillo como en otras máquinas, si es posible desde el BASIC y puede proporcionar resultados tan profesionales como cualquier juego. Sólo hay que aprender poco a poco y saber manejar unas posibilidades que no por desconocidas son inexistentes.

JUGANDO CON LOS CARACTERES

El método más sencillo de hacer moverse objetos en la pantalla es usarlos como si fuesen caracteres, pero esto son limitados y normalmente no son los dibujos que queremos usar (¿alguien ha visto un marciano con forma de "A" o de "K"?) pero estos problemas son fácilmente solucionables. En primer lugar veamos como cambiar el color del fondo y del borde. Basta escribir POKE 53280, X; siendo X un número del cero al quince para que el fondo cambie al color elegido. La misma operación podemos hacer con POKE 53281, Y; siendo Y el número del color que queremos dar al fondo de la pantalla central. Este número, del 0 al 15, se corresponde con uno de los dieciséis que se pueden generar también desde el teclado (pero no para el fondo sino para la letra). El valor de X o de Y correspondiente y su color asociado se da en la tabla 1.

A continuación hemos de definirnos nuestros caracteres especiales, modificando los originales del **Commodore**, éstos están almacenados en una ROM (memoria de solo lectura) de donde los coge el VIC-II, que es la circuito encargado de la pantalla. Para poder crear nuestros propios caracteres debemos hacer lo siguiente: copiar los caracteres en RAM (memoria de lectura y escritura), mo-

Tabla 1.

0: NEGRO	8: NARANJA
1: BLANCO	9: MARRON
2: ROJO	10: ROJO CLARO
3: CIAN	11: GRIS 1
4: PURPURA	12: GRIS 2
5: VERDE	13: VERDE CLARO
6: AZUL	14: AZUL CLARO
7: AMARILLO	15: GRIS 3

dificar los que queramos y dejar los demás como estaban originalmente y por último decirle al VIC-II donde está el nuevo conjunto de caracteres. Para copiarlos deberemos hacer la rutina de la figura 1.

En la línea 10 reservamos espacio en la parte superior de la memoria para que nuestros caracteres puedan meterse en esas direcciones sin ser destruidos por el programa BASIC. En las líneas 20 y 30 activamos la memoria ROM correspondiente al generador de caracteres para realizar la copia (debido a su complejidad no se puede explicar más a fondo este tema, pero es imprescindible) y por último en la línea 40 copiamos el juego de caracteres a la nueva memoria. Las dos últimas líneas (50 y 60) sirven para dejar la máquina en orden, de modo que no se "cuelgue" al terminar el programa. Esta rutina no necesita comprenderse en profundidad, basta con incluirla al principio de nuestros programas y luego seguir trabajando (para aquellos masoquistas que deseen ver su funcionamiento exacto se aconseja que se lean la "Guía de Referencia del Commodore 64" que vende la casa fabricante). Ahora tenemos nuestro juego de caracteres en RAM y podemos modificarlo. Cada carácter tiene un número asociado, que es el que usa la máquina para representarlo internamente.

Para ver el conjunto de los 255 caracteres posible teclee el programa de la figura 2.

En la pantalla aparecerá el código y la representación de todos los caracteres. Si ahora queremos modificar

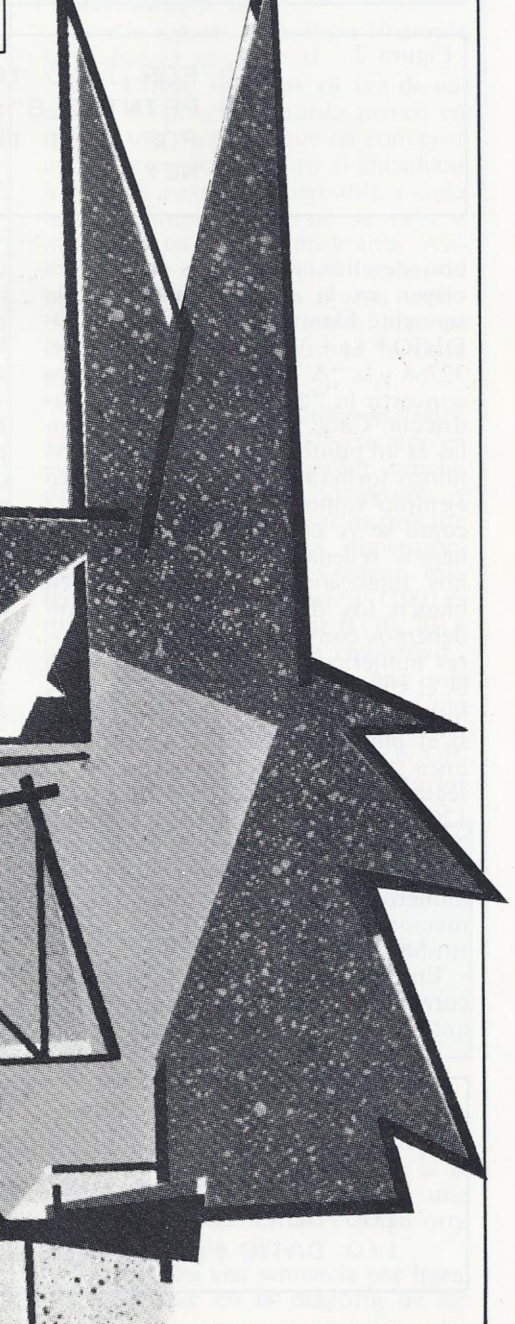
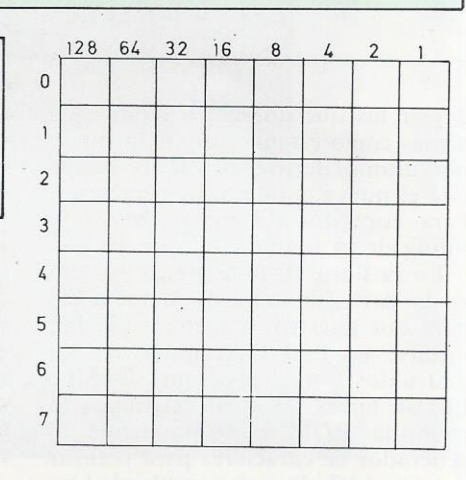


Figura 1.

```
10 POKE 52,48: POKE 56,48: CLR
20 POKE 56334,PEEK(56334) AND 254
30 POKE 1,PEEK(1) AND 251
40 FOR I=0 TO 2047: POKE I+12288,PEEK(I+53248): NEXT I
50 POKE 1,PEEK(1) OR 4
60 POKE 56334,PEEK(56334) OR 1
```

Figura 2.

```
10 FOR I= 0 TO 255
20 PRINT"CLS";I:POKE 1504,I
30 FOR J= 1 TO 500: NEXT J
40 NEXT I
```



uno de ellos, deberemos calcular su origen en la memoria mediante la siguiente fórmula: ORIGEN = (CODIGO * 8) + 32768, así la "@" será el 32768 y la "A" el 32776. Si queremos convertir la "@" utilizaremos la cuadrícula. Cada una de estas cuadrículas es un punto en la pantalla y los 64 juntos forman el caracter, en nuestro ejemplo vamos a dibujar una pelota como se ve en la figura 4. Para ello hemos rellenado los cuadrados que nos interesan y hemos dejado en blanco los demás. A continuación debemos convertir el dibujo en valores numéricos, el método de hacerlo es el siguiente: en cada fila sumamos los valores dados en la parte superior si el punto está relleno. La primera línea sería: 4 + 8 + 16 + 32 = 60, la segunda 2 + 4 + 8 + 16 + 32 + 64 = 126 y las cuatro siguientes valdrían 255, siendo la séptima 126 y la octava 60. Una vez que tenemos convertida a números debemos introducirlos en memoria para lo que utilizaremos la rutina de la figura 5.

En este ejemplo fila cero, fila uno, corresponden a los valores hallados antes. Para introducir nuestra pelota en memoria teclearemos esta rutina adaptada debajo de la que dimos al principio que se encargaba de mover los caracteres, quedando como se ve en la figura 6.

Con lo que habremos modificado la "@" en RAM, del mismo modo podríamos modificar los caracteres que nos interesen. Pero no hemos acabado todavía, si tecleamos este símbolo veremos que en pantalla aparece como siempre, nos falta decirle al VIC-II que deje de buscar los caracteres en ROM y los busque en RAM, para ello usaremos algunas instrucciones mágicas más:

120 POKE 53272, (PEEK (53272) AND 240) + 12

Si ejecuta todo el programa y luego pulsa el símbolo "@" en el teclado verá como aparece nuestra ya lista para jugar. Hemos dado nuestro primer paso en el mundo de los gráficos.

Ahora que ya tenemos unos principios en el manejo de los gráficos, podemos usarlos en cualquiera de los juegos que hemos visto, podemos hacer un ajedrez con fichas que se vean, una aventura que nos dibuje las habitaciones y sus objetos, etc.

Pero quizás su principal aplicación sea en los "arcade" o juegos de acción, capítulo que tocaremos a continuación.

JOYSTICK

El **Commodore 64** tiene dos conectores denominados PORT 1 y PORT 2 en el lateral derecho que, entre otras cosas sirve para conectar *joysticks*, los mandos ideales de juego. El sistema en que se basan para funcionar es muy sencillo, cada una de las direcciones de movimiento dispone de un interruptor que se conecta o se desconecta, según la posición del *joystick*. En total son 5 interruptores (cuatro

Figura 5.

```
70 FOR I= 0 TO 7
80 READ A
90 POKE ORIGEN + I, A
100 NEXT I
110 DATA fila0, fila1, fila2, fila3, fila4, fila5, fila6, fila7
```

de direcciones y 1 de disparo) los que dispone el *joystick*, cada interruptor está numerado (ver figura 7) y su valor se puede leer mediante las dos siguientes instrucciones:

1000 A = PEEK (56320)

1010 A = (NOT (A)) AND 31

El valor 56320 corresponde al *port* 2. Después de ejecutar estas instrucciones dispondremos de un número

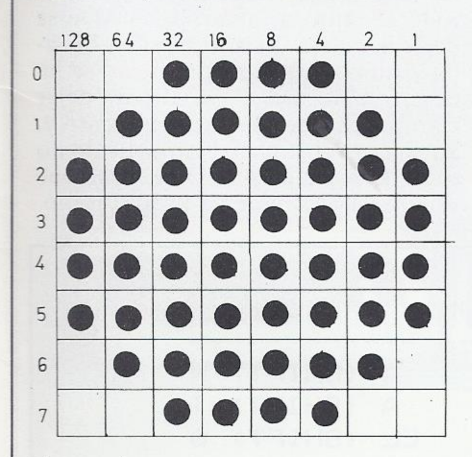


Figura 4.

que nos indicará el estado de los cinco interruptores a la vez, para saber el estado de uno específico deberemos hacer: A AND interruptor, dándonos un cero si el interruptor no está pulsado y el número del interruptor sí está pulsado. Un ejemplo de uso serían las instrucciones de la figura 8.

En las que por cada posición se realizaría una cosa. Ha de tenerse en cuenta, además, que puede haber varios pulsadores apretados a la vez, por ejemplo si disparamos y a la vez movemos a la izquierda y arriba, tendremos pulsados los números 1, 3 y 5.

AGILIZANDO EL PROGRAMA

En los *"Arcade"* uno de los principales problemas reside en la velocidad. Aunque el ordenador es muy rápido, a veces no tiene la velocidad suficiente como para ejecutar todas las instrucciones y, además ser rápido y dar la impresión de movimiento. En

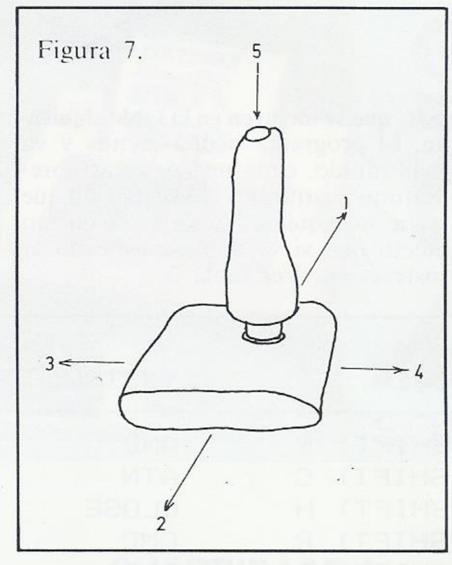


Figura 7.

este apartado daremos algunos consejos para que su programa BASIC vaya más rápido:

a) **Ponga las subrutinas de uso frecuente al principio del programa.** Cuando el ordenador se encuentra un GOTO o un GOSUB salta al principio del programa y empieza a buscar la sentencia que se le ha dicho desde ese punto. Si está colocado al principio, la encuentra antes y tarde menos. Es conveniente colocar al principio del programa una sentencia del tipo:

1 GOTO 5000.

Realizándose a partir de la línea 5000 la inicialización de todos los datos y cuando haya acabado con esto ejecutar

6000 GOTO 10.

Siendo a partir de la línea 10 donde se realice el juego en sí.

b) **Utilice variables en vez de números.** El programa tarda menos en buscar una variable que en convertir un número, por lo tanto al inicializar los datos asigne una variable a cada número, aunque éste no se vaya a modificar en todo el programa. Asimismo, si dos de éstos tienen el mismo valor utilice la misma variable. Aunque esto dificulta algo la comprensión del programa lo hace más rápido.

c) **A ser posible no utilice matrices.** Las matrices tardan mucho tiempo en evaluarse, haciendo perder mucho tiempo. Utilícelas lo menos posible, y en caso de tener que usarlas procure que sean de una sola dimensión, cuantas más dimensiones tengan, más lentas serán. Así la matriz F (100) es más rápida que la matriz B (10, 10) y las dos tienen la misma capacidad.

d) **Utilice variables de una sola letra.** Cuantas más letras tiene una variable, más lento es su manejo. La variable W es más rápida que la OP.

e) **Quite todos los comentarios (REM) del programa.** Los comentarios son muy útiles para comprender como funciona, pero estorban a la máquina. En todo caso guarde una versión sin comentarios y tenga otra (la que use) sin ellos.

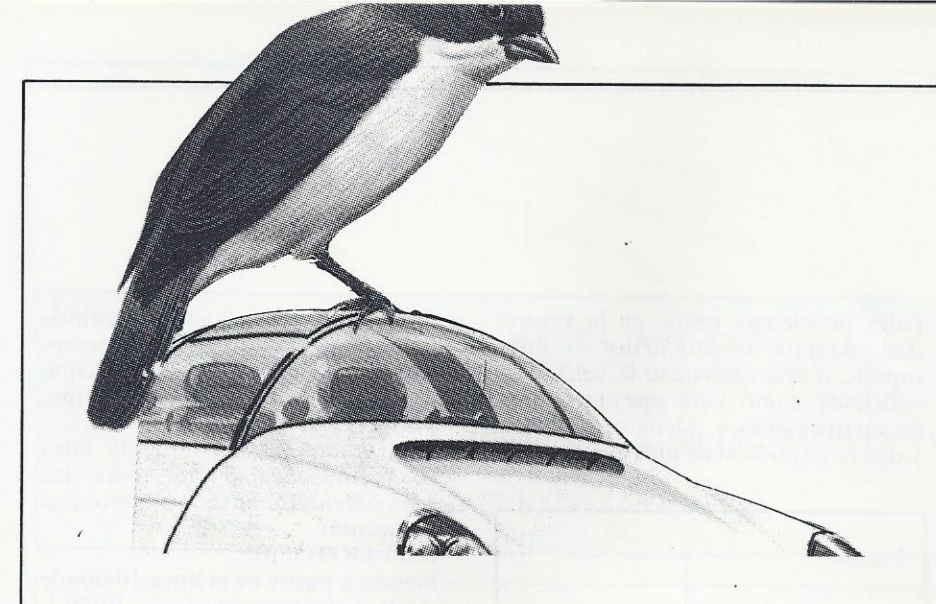
f) **Escriba una sentencia por línea.** Al revés que en la mayoría de los

Figura 8.

```
2000 IF A AND 1 THEN ..(joystick arriba)
2010 IF A AND 2 THEN ..(joystick abajo)
2020 IF A AND 4 THEN ..(joystick izquierda)
2030 IF A AND 8 THEN ..(joystick derecha)
2040 IF A AND 16 THEN . (joystick disparado)
```

Figura 6.

```
70 FOR I= 0 TO 7
80 READ A
90 POKE 12288+I,A
100 NEXT I
110 DATA 60,126,255,255,255,255,126,60
```



ordenadores, el escribir una orden en cada línea hace que el programa sea más rápido que si se escriben varias en la misma y separadas por dos puntos. (Si alguien no se lo cree que lo compruebe).

g) **Utilice las abreviaturas del BASIC que se incluyen en la tabla siguiente.** El programa ocupa menos y va más rápido, cada una de estas abreviaturas sustituye a la instrucción que va a su izquierda y tiene el mismo efecto que si se hubiese tecleado la instrucción. Ver tabla 2.

UN PROGRAMA DE EJEMPLO

En la figura 7 vemos el listado de un programa basado en estas ideas, el jugador controla una raqueta situada en la línea inferior y da a una pelota que rebota contra ella y contra una pared superior de color azul, cada vez que da en ésta el punto en el que impacta se pone de color amarillo. Nuestra misión consiste en cambiar todo el muro al amarillo sin que se cuele la pelota. Esta idea está esbozada en unas líneas generales, no se ha desarrollado más el juego para dejar clara la idea de su funcionamiento y, además, el objetivo de este artículo no es el de listar juegos, si no el de que usted el lector cree su propio juego.

INSTRUCCION	ABREV.	INSTRUCCION	ABREV.
ABS	A (SHIFT) B	AND	A (SHIFT) N
ASC	A (SHIFT) C	ATN	A (SHIFT) T
CHR$	C (SHIFT) H	CLOSE	CL (SHIFT) O
CLR	C (SHIFT) R	CMD	C (SHIFT) M
CONT	C (SHIFT) O	DATA	D (SHIFT) A
DEF	D (SHIFT) E	DIM	D (CHIFT) I
END	E (SHIFT) N	EXP	E (CHIFT) X
FOR	F (SHIFT) O	FRE	F (SHIFT) R
GET	G (SHIFT) E	GOSUB	GO (SHIFT) S
GOTO	G (SHIFT) O	INPUT	I (SHIFT) N
LEFT$	LE (SHIFT) F	LET	L (SHIFT) E
LIST	L (SHIFT) I	LOAD	L (SHIFT) O
MID$	M (SHIFT) I	NEXT	N (SHIFT) E
NOT	N (SHIFT) O	OPEN	O (SHIFT) P
PEEK	P (SHIFT) E	POKE	P (SHIFT) O
PRINT	?	PRINT#	P (SHIFT) R
READ	R (SHIFT) E	RESTORE	RE (SHIFT) S
RETURN	RE (SHIFT) T	RIGHT$	R (SHIFT) I
RND	R (SHIFT) N	RUN	R (SHIFT) U
SAVE	S (SHIFT) A	SGN	S (SHIFT) G
SIN	S (SHIFT) I	SPC(	S (SHIFT) P
SQR	S (SHIFT) Q	STEP	ST (SHIFT) E
STOP	S (SHIFT) T	STR$	ST (SHIFT) R
SYS	S (SHIFT) Y	TAB(	T (SHIFT) A
THEN	T (SHIFT) H	TIME	TI
TIME$	TI$	USR	U (SHIFT) S
VAL	V (SHIFT) A	VERIFY	V (SHIFT) E
WAIT	W (SHIFT) A		

SUSCRIBASE A

COMMODORE MAGAZINE

(12 NUMEROS)

TARIFA DE PRECIOS DE SUSCRIPCION

	CORREO ORDINARIO		CORREO CERTIFICADO		CORREO AEREO		CORREO AEREO-CERTIF.	
	PTAS.	$	PTAS.	$	PTAS.	$	PTAS.	$
ESPAÑA	3.000	21	3.273	23	3.055	22	3.333	24
EUROPA, MARRUECOS, TUNEZ, TURQUIA, ARGELIA Y CHIPRE .	3.456	25	4.272	31	3.600	26	4.418	31
COSTA RICA, CUBA, CHILE, PARAGUAY Y REP. DOMINICANA .	3.396	24	4.212	30	4.164	30	4.980	36
GIBRALTAR Y PORTUGAL	3.264	23	4.080	29	3.149	22	3.965	28
FILIPINAS	3.264	23	3.540	25	3.775	27	4.050	29
RESTO DEL MUNDO	3.456	25	4.272	31	4.224	30	5.040	36

CUPON DE PEDIDO

Recorte y envíe este cupón a: **COMMODORE MAGAZINE,** C/Bravo Murillo, 377, 5ª. A - Madrid-20.

El importe lo abonaré: POR CHEQUE ☐ CONTRA REEMBOLSO ☐

CON TARJETA DE CREDITO ☐ American Express ☐ Visa ☐ Interbank ☐

Número de mi Tarjeta: ________________ Fecha de caducidad: ________

NOMBRE ________________________________

DIRECCION ________________________________

CIUDAD ________________ D.P. ____ PROVINCIA ________________

CUANDO SE TIENE UN COMMODORE 64 ES MUY DIFICIL SER MODESTO

Cuando se tiene un ordenador personal con 64K de memoria, una magnífica resolución, 16 colores, efectos tridimensionales con "sprites", un sonido equivalente al de un sintetizador, un teclado profesional con 62 caracteres gráficos, toda una amplia gama de periféricos profesionales, la más completa serie de programas educativos, profesionales y de video-juegos...; en resumen, cuando se tiene un ordenador personal como no existe ningún otro en el mercado y el más vendido mundialmente, cuando se tiene el Commodore 64, es muy difícil mostrarlo sin que el orgullo se te note.



EL ORDENADOR PERSONAL DE LA FAMILIA MAS POTENTE

– Sistemas de gestión profesionales series 8000 Y 700. – Ordenador portátil SX 64.
– Ordenador personal COMMODORE 64. – Ordenador familiar VIC 20.



MICROELECTRONICA Y CONTROL, S.A.
c/ Taquígrafo Serra, 7, 5.° BARCELONA-29 c/ Princesa, 47, 3.° G MADRID-8

SLOGAN